

Frontiers of Multimedia Research

ACM Books

Editor in Chief

M. Tamer Özsu, *University of Waterloo*

ACM Books is a new series of high-quality books for the computer science community, published by ACM in collaboration with Morgan & Claypool Publishers. ACM Books publications are widely distributed in both print and digital formats through booksellers and to libraries (and library consortia) and individual ACM members via the ACM Digital Library platform.

Frontiers of Multimedia Research

Editor: Shih-Fu Chang, *Columbia University*
2018

The Continuing Arms Race: Code-Reuse Attacks and Defenses

Editors: Thorsten Holz, *Ruhr-Universität Bochum*
Per Larsen, *Immunant, Inc.*
Ahmad-Reza Sadeghi, *Technische Universität Darmstadt*
2018

Shared-Memory Parallelism Can Be Simple, Fast, and Scalable

Julian Shun, *University of California, Berkeley*
2017

Computational Prediction of Protein Complexes from Protein Interaction Networks

Sriganesh Srihari, *The University of Queensland Institute for Molecular Bioscience*
Chern Han Yong, *Duke-National University of Singapore Medical School*
Limsoon Wong, *National University of Singapore*
2017

The Handbook of Multimodal-Multisensor Interfaces, Volume 1: Foundations, User Modeling, and Common Modality Combinations

Editors: Sharon Oviatt, *Incaa Designs*
Björn Schuller, *University of Passau and Imperial College London*
Philip R. Cohen, *Voicebox Technologies*
Daniel Sonntag, *German Research Center for Artificial Intelligence (DFKI)*
Gerasimos Potamianos, *University of Thessaly*
Antonio Krüger, *German Research Center for Artificial Intelligence (DFKI)*
2017

Communities of Computing: Computer Science and Society in the ACM

Thomas J. Misa, Editor, *University of Minnesota*
2017

Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining

ChengXiang Zhai, *University of Illinois at Urbana-Champaign*
Sean Massung, *University of Illinois at Urbana-Champaign*
2016

An Architecture for Fast and General Data Processing on Large Clusters

Matei Zaharia, *Stanford University*
2016

Reactive Internet Programming: State Chart XML in Action

Franck Barbier, *University of Pau, France*
2016

Verified Functional Programming in Agda

Aaron Stump, *The University of Iowa*
2016

The VR Book: Human-Centered Design for Virtual Reality

Jason Jerald, *NextGen Interactions*
2016

Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age

Robin Hammerman, *Stevens Institute of Technology*
Andrew L. Russell, *Stevens Institute of Technology*
2016

Edmund Berkeley and the Social Responsibility of Computer Professionals

Bernadette Longo, *New Jersey Institute of Technology*
2015

Candidate Multilinear Maps

Sanjam Garg, *University of California, Berkeley*
2015

Smarter Than Their Machines: Oral Histories of Pioneers in Interactive Computing

John Cullinane, *Northeastern University; Mossavar-Rahmani Center for Business and Government, John F. Kennedy School of Government, Harvard University*
2015

A Framework for Scientific Discovery through Video Games

Seth Cooper, *University of Washington*
2014

Trust Extension as a Mechanism for Secure Code Execution on Commodity Computers

Bryan Jeffrey Parno, *Microsoft Research*
2014

Embracing Interference in Wireless Systems

Shyamnath Gollakota, *University of Washington*
2014

Frontiers of Multimedia Research

Shih-Fu Chang

Columbia University

ACM Books #17



Copyright © 2018 by the Association for Computing Machinery
and Morgan & Claypool Publishers

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopy, recording, or any other except for brief quotations in printed reviews—without the prior permission of the publisher.

Designations used by companies to distinguish their products are often claimed as trademarks or registered trademarks. In all instances in which Morgan & Claypool is aware of a claim, the product names appear in initial capital or all capital letters. Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

Frontiers of Multimedia Research

Shih-Fu Chang, editor

books.acm.org

www.morganclaypoolpublishers.com

ISBN: 978-1-97000-107-5 hardcover

ISBN: 978-1-97000-104-4 paperback

ISBN: 978-1-97000-105-1 eBook

ISBN: 978-1-97000-106-8 ePub

Series ISSN: 2374-6769 print 2374-6777 electronic

DOIs:

10.1145/3122865 Book	10.1145/3122865.3122872 Chapter 6
10.1145/3122865.3122866 Preface	10.1145/3122865.3122873 Chapter 7
10.1145/3122865.3122867 Chapter 1	10.1145/3122865.3122874 Chapter 8
10.1145/3122865.3122868 Chapter 2	10.1145/3122865.3122875 Chapter 9
10.1145/3122865.3122869 Chapter 3	10.1145/3122865.3122876 Chapter 10
10.1145/3122865.3122870 Chapter 4	10.1145/3122865.3122877 Chapter 11
10.1145/3122865.3122871 Chapter 5	10.1145/3122865.3122878 References

A publication in the ACM Books series, #17

Editor in Chief: M. Tamer Özsu, *University of Waterloo*

First Edition

10 9 8 7 6 5 4 3 2 1

Contents

Preface xi

PART I MULTIMEDIA CONTENT ANALYSIS 1

Chapter 1 Deep Learning for Video Classification and Captioning 3

Zuxuan Wu, Ting Yao, Yanwei Fu, Yu-Gang Jiang

- 1.1 Introduction 3
- 1.2 Basic Deep Learning Modules 4
- 1.3 Video Classification 8
- 1.4 Video Captioning 14
- 1.5 Benchmarks and Challenges 19
- 1.6 Conclusion 29

Chapter 2 Audition for Multimedia Computing 31

Gerald Friedland, Paris Smaragdis, Josh McDermott, Bhiksha Raj

- 2.1 A New Field 32
- 2.2 Background 35
- 2.3 Data for Computer Audition 39
- 2.4 The Nature of Audio Data 40
- 2.5 Dealing with the Peculiarities of Sound 41
- 2.6 Potential Applications 49
- 2.7 Conclusion 49
- Acknowledgments 50

Chapter 3 Multimodal Analysis of Free-standing Conversational Groups 51

Xavier Alameda-Pineda, Elisa Ricci, Nicu Sebe

- 3.1 Introduction 52
- 3.2 Related Work 55
- 3.3 The SALSA Dataset 58
- 3.4 Matrix Completion for Multimodal Pose Estimation 59
- 3.5 Experiments 66
- 3.6 Conclusion 73

Chapter 4 Encrypted Domain Multimedia Content Analysis 75

Pradeep K. Atrey, Ankita Lathey, Abukari M. Yakubu

- 4.1 Introduction 75
- 4.2 SPED: An Overview 78
- 4.3 Image Processing in Encrypted Domain 84
- 4.4 Video Processing in Encrypted Domain 91
- 4.5 Audio Processing in Encrypted Domain 95
- 4.6 Further Discussion 101
- 4.7 Conclusion 104

Chapter 5 Efficient Similarity Search 105

Hervé Jegou

- 5.1 Background 106
- 5.2 Cell-probe Algorithms 113
- 5.3 Sketches and Binary Embeddings 120
- 5.4 Searching and Similarity Estimation with Quantization 127
- 5.5 Hybrid Approaches: The Best of Probing and Sketching 131
- 5.6 Searching for Non-Euclidean Metrics and Graph-based Approaches 132
- 5.7 Conclusion 134

PART II HUMAN-CENTERED MULTIMEDIA COMPUTING 135

Chapter 6 Social-Sensed Multimedia Computing 137

Peng Cui

- 6.1 Semantic Gap vs. Need Gap 139
- 6.2 Social-Sensed Multimedia Computing 142

- 6.3 Basic Problems and Key Technologies 144
- 6.4 Recent Advances 146
- 6.5 Exemplary Applications 150
- 6.6 Discussions on Future Directions 153
- 6.7 Conclusion 157

Chapter 7 Situation Recognition Using Multimodal Data 159

Vivek Singh

- 7.1 The Emerging Eco-system and a Motivating Application 162
- 7.2 Defining *Situation* 164
- 7.3 A Framework for Situation Recognition 170
- 7.4 EventShop: A Toolkit for Situation Recognition 177
- 7.5 Building Situation-Aware Applications Using EventShop 182
- 7.6 Open Challenges and Opportunities 185
- 7.7 Conclusion 188
- Acknowledgments 189

Chapter 8 Hawkes Processes for Events in Social Media 191

Marian-Andrei RizoIU, Young Lee, Swapnil Mishra, Lexing Xie

- 8.1 Introduction 191
- 8.2 Preliminary: Poisson Processes 193
- 8.3 Hawkes Processes 197
- 8.4 Simulating Events from Hawkes Processes 202
- 8.5 Estimation of Hawkes Processes Parameters 205
- 8.6 Constructing a Hawkes Model for Social Media 209
- 8.7 Conclusion 217

Chapter 9 Utilizing Implicit User Cues for Multimedia Analytics 219

Subramanian Ramanathan, Syed Omer Gilani, Nicu Sebe

- 9.1 Introduction 219
- 9.2 Inferring Scene Semantics from Eye Movements 222
- 9.3 Eye Fixations as Implicit Annotations for Object Recognition 226
- 9.4 Emotion and Personality Type Recognition via Physiological Signals 236
- 9.5 Conclusion 250

PART III MULTIMEDIA COMMUNICATION AND SYSTEMS 253

Chapter 10 Multimedia Fog Computing: Minions in the Cloud and Crowd 255

*Cheng-Hsin Hsu, Hua-Jun Hong, Tarek Elgamal, Klara Nahrstedt,
Nalini Venkatasubramanian*

- 10.1 Introduction 255
- 10.2 Related Work 258
- 10.3 Challenges 260
- 10.4 Distributed Multimedia Applications: What We Can Learn from Prior Studies 263
- 10.5 Deployment: Open-Source Platforms 280
- 10.6 Conclusion 285

Chapter 11 Cloud Gaming 287

*Kuan-Ta Chen, Wei Cai, Ryan Shea, Chun-Ying Huang, Jiangchuan Liu,
Victor C. M. Leung, Cheng-Hsin Hsu*

- 11.1 Overview on Cloud Gaming Research 289
- 11.2 GamingAnywhere: An Open-Source Cloud Gaming Platform 291
- 11.3 Cloud Deployment 298
- 11.4 Thin Client Design 302
- 11.5 Communication 307
- 11.6 Future Paradigm of Cloud Gaming 310
- 11.7 Conclusion 314

Bibliography 315

Index 379

Editor Biography 399

Preface

The field of multimedia is dedicated to research and studies that leverage multiple modalities of signals and data in developing intelligent systems and technologies. Be it search engine, recommendation system, streaming service, interactive agent, or collaborative system, multimedia plays a critical role in ensuring full understanding of multimodal sensory signals, robust modeling of user-content interaction, natural and rich communication experience, and scalable system deployment. The goal is to utilize unique contributions from each modality, integrate complementary synergies, and achieve the best performance and novel functions beyond what's separately available in each individual medium. In this community, most contributors also maintain strong activities in other disciplines such as networking, computer vision, human-computer interaction, and machine learning. But the field of multimedia is unique in offering a rich and dynamic forum for researchers from "traditional" fields to collaborate and develop new solutions and knowledge that transcend the boundaries of individual disciplines.

The field enjoys a long history of vibrant research. For example, the flagship ACM SIGMM Multimedia Conference was established in 1993, celebrating its 25th anniversary this year. The community also has several well-known conferences and journals organized by ACM, IEEE, and other groups, attracting a large number of researchers and practitioners from around the world. However, despite the prolific research activities and outcomes, there has been less effort toward developing books that serve as an introduction to the rich spectrum of topics in this broad field. Most of the few books available today either focus on specific subfields or basic background. There is a lack of tutorial-style materials covering the active topics being pursued by the leading researchers at frontiers of the field.

SIGMM launched a new initiative to address this need in 2015, by selecting and inviting 12 rising-star speakers from different subfields of multimedia to deliver plenary tutorial style talks at ACM Multimedia 2015. Each speaker discussed challenges and the state of the art within their prospective research areas in a

general manner to the broad community. Topics covered were comprehensive, including multimedia content understanding, multimodal human-human and human-computer interaction, multimedia social media, and multimedia system architecture and deployment. Following the very positive responses to the talks, these rising-star speakers were invited to expand the content covered in their talks to chapters that can be used as reference materials for researchers, students, and practitioners. Each resulting chapter discusses problems, technical challenges, state-of-the-art approaches and performances, open issues, and promising directions for future work. Collectively, the chapters provide an excellent sampling of major topics addressed by the community as a whole. This book, capturing outcomes of such efforts, is well positioned to fill the aforementioned needs by providing tutorial-style reference materials for frontier topics of multimedia.

Section 1 of the book includes five chapters that are focused on analysis and understanding of multimedia content. Topics covered range from analysis of video content, audio content, multimodal content about interaction of freestanding conversational groups, and analysis of multimedia data in the encrypted format for preserving privacy on cloud servers, to efficient approximate similarity search techniques for searching over large-scale databases.

First, Zuxuan Wu et al. review current research on understanding video content by detecting the classes of actions or events contained in a given video clip and generation of full-sentence captions describing the content in each such video. Unlike previous surveys, this review focuses on solutions based on deep learning, reflecting the recent trend of research in this area. The chapter also gives extensive reviews of the datasets used in state-of-the-art research and benchmarking efforts.

Extending the modality from video to audio, in Chapter 2, Gerald Friedland et al. introduce the field of computer audition, aiming to develop the theory behind artificial systems that can extract information from sound. This chapter reviews the research datasets available, appropriate representations needed for audio, and a few challenging problems such as automatic extraction of hierarchical semantic structures from audio content and automatic discovery of high-level semantic concepts from massive audio data and associated metadata.

The holy grail of research for the multimedia community is to be able to integrate and fuse information extracted from multiple modalities of data. In Chapter 3, Xavier Alameda-Pineda et al. present an excellent example and emergent research challenges in the application of detecting social interaction among freestanding conversational groups. The chapter includes overviews of research issues, approaches, evaluation of joint estimation of head and body poses using multi-

modality data (such as wearable sensors and distributed camera networks), and results of detecting dynamic group formation of interacting people.

Chapter 4 addresses a novel emerging topic prompted by the popular approach to multimedia analysis using cloud computing servers. When multimedia data is sent to the cloud for storage or processing, there is a risk of privacy breach via unauthorized access by third parties to the content in the cloud. Pradeep Atrey et al. review state-of-the-art methods and open issues for processing multimedia content in the encrypted domain without needing to convert data to the original format. This allows content to stay in its protected form while useful analysis is performed on it.

In Chapter 5, Hervé Jégou surveys efficient techniques for finding approximate solutions for similarity search, which is of particular interest when searching massive multimedia data like images, videos, and audio recordings. Jégou considers various performance factors like query speed, memory requirement, and search accuracy. Multiple frameworks based on locality sensitive hashing (LSH), quantization/compression, and hybrid combinations are also reviewed in a coherent manner.

In Section 2 of the book, the emphasis shifts from content analysis to human-centered aspects of multimedia computing. This new focus goes beyond extraction of semantic information from multimedia data. Instead, the broad research scope incorporates understanding of users and user-content interaction so as to improve effectiveness of multimedia systems in many applications, such as search and recommendation.

Under the human-centric theme, Chapter 6, authored by Peng Cui, discusses the evolution of multimedia computing paradigms from the data-centric, to the content-centric, and recently to the human-centric. Cui presents a new framework, called social-sensed multimedia computing, to capture many key issues involved and advances achieved, including understanding of user-content interaction behavior, understanding of user intent, multimedia representation considering user intention, and integration of heterogeneous data sensed on multimedia social networks.

Chapter 7 follows the human-centric theme and further moves the focus from processing individual multimedia data streams to processing a large number of heterogeneous streams in different modalities involving a large number of people. Analysis of such massive streams offers the possibility of detecting important situations of society, such as socio-economic affairs, as well as the living environment. Vivek Singh provides an overview of the problem definition, research framework,

and the EventShop toolkit he developed for application development in this emerging area.

The extension to the human-centric computing paradigm also calls for formal mathematical theories and tools for explaining the phenomena observed, such as the information propagation behaviors and the occurrences of information cascades on social networks. In Chapter 8, Marian-Andrei RizoIU et al. review stochastic processes such as the Hawkes point process for modeling discrete, interdependent events over continuous time. These are strongly related to patterns corresponding to retweet cascade events on social media. Successful models like these can help researchers understand information dissemination patterns and predict popularity on social media.

Interaction between users and content reveals not only the intent of the user (covered in Chapter 6), but also attributes of the content as well as of the user him/herself. Such interaction can be manifested in multiple forms including explicit cues such as visual and verbal expressions, and implicit cues such as eye movement and physiological signals like brain activity and heart rate. Chapter 9 includes a survey by Subramanian Ramanathan et al. on how such implicit user interaction cues can be explored to improve analysis of content (scene understanding) and user (user emotion recognition).

To support research and development of emerging multimedia topics discussed above, there is a critical need for new generations of communication and computing systems that take into account the unique requirements of multimedia, such as real-time, high bandwidth, distributiveness, major power consumption, and resource uncertainty. The popular cloud-based computing systems, though prevalent for many applications, are not suitable for large-scale multimedia applications such as cloud-based gaming service and animation rendering service.

The last section of the book focuses on the systems aspect, covering distinct topics of multimedia fog computing (Chapter 10) and cloud gaming (Chapter 11). Cheng-Hsin Hsu et al. survey the emerging paradigm focused on fog computing, in which computing services are crowdsourced to the edge nodes or even to the client devices on the user end. This offers major potential benefits in terms of low latency, location awareness, scalability, and heterogeneity. However, it also poses many significant challenges in areas such as resource discovery, resource allocation and management, quality of service, and security. Discussion of these challenges, along with recent advances in this area, are presented in Chapter 10.

Finally, as a concrete example of large-scale distributed multimedia computing systems, Chapter 11 (by Kuan-Ta Chen et al.) presents a comprehensive survey of cloud gaming, with emphasis on the development of platform and testbed, test

scenarios, and evaluation of performance, in order to enhance optimal design of various components of the complex cloud gaming systems. In particular, the chapter overviews extensive research in areas such as open-source platforms, cloud deployment, client design, and communication between gaming servers and clients.

The scope of this book is by no means exhaustive or complete. For example, it can be expanded to include other important topics such as (but not limited to) multimedia content generation, multimodal knowledge discovery and representation, multimedia immersive networked environments, and applications in areas like healthcare, learning, and infrastructure. Nonetheless, the comprehensive survey materials already covered in the book provide an excellent foundation for exploring additional topics mentioned above, and many other relevant fields.

We would like to give sincere acknowledgment to Dr. Svebor Karaman, who has provided tremendous assistance in communicating with contributors and organizing the content of this book. In addition, Diane Cerra and her team at Morgan & Claypool Publishers have provided valuable guidance and editorial help.



PART

**MULTIMEDIA
CONTENT
ANALYSIS**

1 Deep Learning for Video Classification and Captioning

Zuxuan Wu (University of Maryland, College Park),
Ting Yao (Microsoft Research Asia),
Yanwei Fu (Fudan University),
Yu-Gang Jiang (Fudan University)

1.1 Introduction

Today's digital contents are inherently multimedia: text, audio, image, video, and so on. Video, in particular, has become a new way of communication between Internet users with the proliferation of sensor-rich mobile devices. Accelerated by the tremendous increase in Internet bandwidth and storage space, video data has been generated, published, and spread explosively, becoming an indispensable part of today's big data. This has encouraged the development of advanced techniques for a broad range of video understanding applications including online advertising, video retrieval, video surveillance, etc. A fundamental issue that underlies the success of these technological advances is the understanding of video contents. Recent advances in deep learning in image [[Krizhevsky et al. 2012](#), [Russakovsky et al. 2015](#), [Girshick 2015](#), [Long et al. 2015](#)] and speech [[Graves et al. 2013](#), [Hinton et al. 2012](#)] domains have motivated techniques to learn robust video feature representations to effectively exploit abundant multimodal clues in video data.

In this chapter, we review two lines of research aiming to stimulate the comprehension of videos with deep learning: video classification and video captioning. While video classification concentrates on automatically labeling video clips based on their semantic contents like human actions or complex events, video captioning

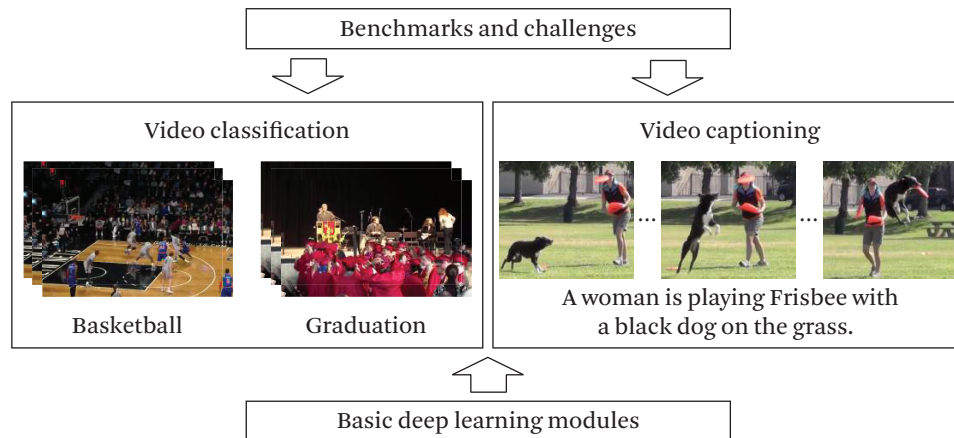


Figure 1.1 An overview of the organization of this chapter.

attempts to generate a complete and natural sentence, enriching video classification’s single label to capture the most informative dynamics in videos.

There have been several efforts surveying the literature on video content understanding. Most of the approaches surveyed in these works adopted handcrafted features coupled with typical machine learning pipelines for action recognition and event detection [Aggarwal and Ryoo 2011, Turaga et al. 2008, Poppe 2010, Jiang et al. 2013]. In contrast, this chapter focuses on discussing state-of-the-art deep learning techniques not only for video classification but also video captioning. As deep learning for video analysis is an emerging and vibrant field, we hope this chapter could help stimulate future research along the line.

Figure 1.1 shows the organization of this chapter. To make it self-contained, we first introduce the basic modules that are widely adopted in state-of-the-art deep learning pipelines in Section 1.2. After that, we discuss representative works on video classification and video captioning in Section 1.3 and Section 1.4, respectively. Finally, in Section 1.5 we provide a review of popular benchmarks and challenges in that are critical for evaluating the technical progress of this vibrant field.

1.2 Basic Deep Learning Modules

In this section, we briefly review basic deep learning modules that have been widely adopted in the literature for video analysis.

1.2.1 Convolutional Neural Networks (CNNs)

Inspired by the visual perception mechanisms of animals [Hubel and Wiesel 1968] and the McCulloch-Pitts model [McCulloch and Pitts 1943], Fukushima proposed the “neocognitron” in 1980, which is the first computational model of using local connectivities between neurons of a hierarchically transformed image [Fukushima 1980]. To obtain the translational invariance, Fukushima applied neurons with the same parameters on patches of the previous layer at different locations; thus this can be considered the predecessor of convolutional neural networks (CNN). Further inspired by this idea, LeCun et al. [1990] designed and trained the modern framework of CNNs LeNet-5, and obtained the state-of-the-art performance on several pattern recognition datasets (e.g., handwritten character recognition). LeNet-5 has multiple layers and is trained with the back-propagation algorithm in an end-to-end formulation, that is, classifying visual patterns directly by using raw images. However, limited by the scale of labeled training data and computational power, LeNet-5 and its variants [LeCun et al. 2001] did not perform well on more complex vision tasks until recently.

To better train deep networks, Hinton et al. in 2006 made a breakthrough and introduced deep belief networks (DBNs) to greedily train each layer of the network in an unsupervised manner. And since then, researchers have developed more methods to overcome the difficulties in training CNN architectures. Particularly, AlexNet, as one of the milestones, was proposed by Krizhevsky et al. in 2012 and was successfully applied to large-scale image classification in the well-known ImageNet Challenge. AlexNet contains five convolutional layers followed by three fully connected (fc) layers [Krizhevsky et al. 2012]. Compared with LeNet-5, two novel components were introduced in AlexNet:

1. ReLUs (Rectified Linear Units) are utilized to replace the tanh units, which makes the training process several times faster.
2. Dropout is introduced and has proven to be very effective in alleviating overfitting.

Inspired by AlexNet, several variants, including VGGNet [Simonyan and Zisserman 2015], GoogLeNet [Szegedy et al. 2015a], and ResNet [He et al. 2016b], have been proposed to further improve the performance of CNNs on visual recognition tasks:

VGGNet has two versions, VGG16 and VGG19, which contain 16 and 19 layers, respectively [Simonyan and Zisserman 2015]. VGGNet pushed the depth of CNN architecture from 8 layers as in AlexNet to 16–19 layers, which greatly

improves the discriminative power. In addition, by using very small (3×3) convolutional filters, VGGNet is capable of capturing details in the input images.

GoogLeNet is inspired by the Hebbian principle with multi-scale processing and it contains 22 layers [Szegedy et al. 2015a]. A novel CNN architecture commonly referred to as Inception is proposed to increase both the depth and the width of CNN while maintaining an affordable computational cost. There are several extensions upon this work, including BN-Inception-V2 [Szegedy et al. 2015b], Inception-V3 [Szegedy et al. 2015b], and Inception-V4 [Szegedy et al. 2017].

ResNet, as one of the latest deep architectures, has remarkably increased the depth of CNN to 152 layers using deep residual layers with skip connections [He et al. 2016b]. ResNet won the first place in the 2015 ImageNet Challenge and has recently been extended to more than 1000 layers on the CIFAR-10 dataset [He et al. 2016a].

From AlexNet, VGGNet, and GoogLeNet to the more recent ResNet, one trend in the evolution of these architectures is to deepen the network. The increased depth allows the network to better approximate the target function, generating better feature representations with higher discriminative power. In addition, various methods and strategies have been proposed from different aspects, including but not limited to Maxout [Goodfellow et al. 2013], DropConnect [Wan et al. 2013], and Batch Normalization [Ioffe and Szegedy 2015], to facilitate the training of deep networks. Please refer to Bengio et al. [2013] and Gu et al. [2016] for a more detailed review.

1.2.2 Recurrent Neural Networks (RNNs)

The CNN architectures discussed above are all feed-forward neural networks (FFNNs) whose connections do not form cycles, which makes them insufficient for sequence labeling. To better explore the temporal information of sequential data, recurrent connection structures have been introduced, leading to the emergence of recurrent neural networks (RNNs). Unlike FFNNs, RNNs allow cyclical connections to form cycles, which thus enables a “memory” of previous inputs to persist in the network’s internal state [Graves 2012]. It has been pointed out that a finite-sized RNN with sigmoid activation functions can simulate a universal Turing machine [Siegelmann and Sontag 1991].

The basic RNN block, at a time step t , accepts an external input vector $\mathbf{x}^{(t)} \in \mathbb{R}^n$ and generates an output vector $\mathbf{z}^{(t)} \in \mathbb{R}^m$ via a sequence of hidden states

$\mathbf{h}^{(t)} \in \mathbb{R}^r$:

$$\mathbf{h}^{(t)} = \sigma \left(W_x \mathbf{x}^{(t)} + W_h \mathbf{h}^{(t-1)} + \mathbf{b}_h \right) \quad (1.1)$$

$$\mathbf{z}^{(t)} = \text{softmax} \left(W_z \mathbf{h}^{(t)} + \mathbf{b}_z \right)$$

where $W_x \in \mathbb{R}^{r \times n}$, $W_h \in \mathbb{R}^{r \times r}$, and $W_z \in \mathbb{R}^{m \times r}$ are weight matrices and \mathbf{b}_h and \mathbf{b}_z are biases. The σ is defined as sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$ and $\text{softmax}(\cdot)$ is the softmax function.

A problem with RNN is that it is not capable of modeling long-range dependencies and is unable to store information about past inputs for a very long period [Bengio et al. 1994], though one large enough RNN should, in principle, be able to approximate the sequences of arbitrary complexity. Specifically, two well-known issues—*vanishing* and *exploding* gradients, exist in training RNNs: the vanishing gradient problem refers to the exponential shrinking of gradients' magnitude as they are propagated back through time; and the exploding gradient problem refers to the explosion of long-term components due to the large increase in the norm of the gradient during training sequences with long-term dependencies. To solve these issues, researchers introduced Long short-term memory models.

Long short-term memory (LSTM) is an RNN variant that was designed to store and access information in a long time sequence. Unlike standard RNNs, non-linear multiplicative gates and a memory cell are introduced. These gates, including input, output, and forget gates, govern the information flow into and out of the memory cell. The structure of an LSTM unit is illustrated in Figure 1.2.

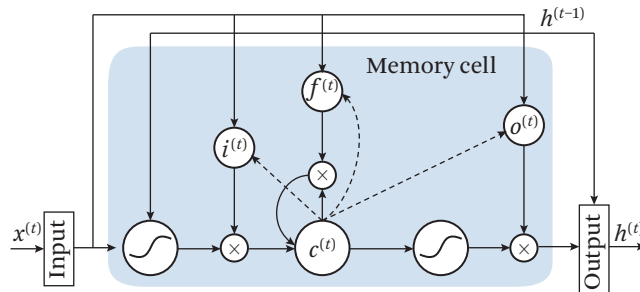


Figure 1.2 The structure of an LSTM unit. (Modified from Wu et al. [2016])

More specifically, given a sequence of an external input vector $\mathbf{x}^{(t)} \in \mathbb{R}^n$, an LSTM maps the input to an output vector $\mathbf{z}^{(t)} \in \mathbb{R}^m$ by computing activations of the units in the network with the following equations recursively from $t = 1$ to $t = T$:

$$\begin{aligned}\mathbf{i}^{(t)} &= \sigma(W_{xi}\mathbf{x}^{(t)} + W_{hi}\mathbf{h}^{(t-1)} + W_{ci}\mathbf{c}^{(t)} + \mathbf{b}_i), \\ \mathbf{f}^{(t)} &= \sigma(W_{xf}\mathbf{x}^{(t)} + W_{hf}\mathbf{h}^{(t)} + W_{cf}\mathbf{c}^{(t)} + \mathbf{b}_f), \\ \mathbf{c}^{(t)} &= \mathbf{f}^{(t)}\mathbf{c}^{(t-1)} + \mathbf{i}_t \tanh(W_{xc}\mathbf{x}^{(t)} + W_{hc}\mathbf{h}^{(t-1)} + \mathbf{b}_c), \\ \mathbf{o}^{(t)} &= \sigma(W_{xo}\mathbf{x}^{(t)} + W_{ho}\mathbf{h}^{(t-1)} + W_{co}\mathbf{c}^{(t)} + \mathbf{b}_o), \\ \mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \tanh(\mathbf{c}^{(t)}),\end{aligned}\tag{1.2}$$

where $\mathbf{x}^{(t)}$, $\mathbf{h}^{(t)}$ are the input and hidden vectors with the subscription t denoting the t -th time step, while $\mathbf{i}^{(t)}$, $\mathbf{f}^{(t)}$, $\mathbf{c}^{(t)}$, $\mathbf{o}^{(t)}$ are, respectively, the activation vectors of the input gate, forget gate, memory cell, and output gate. $W_{\alpha\beta}$ denotes the weight matrix between α and β . For example, the weight matrix from the input $\mathbf{x}^{(t)}$ to the input gate $\mathbf{i}^{(t)}$ is W_{xi} .

In Equation 1.2 and Figure 1.2 and at time step t , the input $\mathbf{x}^{(t)}$ and the previous states $\mathbf{h}^{(t-1)}$ are used as the input of LSTM. The information of the memory cell is updated/controlled from two sources: (1) the previous cell memory unit $\mathbf{c}^{(t-1)}$ and (2) the input gate's activation \mathbf{i}_t . Specifically, $\mathbf{c}^{(t-1)}$ is multiplied by the activation from the forget gate $\mathbf{f}^{(t)}$, which learns to forget the information of the previous states. In contrast, the \mathbf{i}_t is combined with the new input signal to consider new information. LSTM also utilizes the output gate $\mathbf{o}^{(t)}$ to control the information received by hidden state variable $\mathbf{h}^{(t)}$. To sum up, with these explicitly designed memory units and gates, LSTM is able to exploit the long-range temporal memory and avoids the issues of vanishing/exploding gradients. LSTM has recently been popularly used for video analysis, as will be discussed in the following sections.

1.3 Video Classification

The sheer volume of video data has motivated approaches to automatically categorizing video contents according to classes such as human activities and complex events. There is a large body of literature focusing on computing effective local feature descriptors (e.g., HoG, HoF, MBH, etc.) from spatio-temporal volumes to account for temporal clues in videos. These features are then quantized into bag-of-words or Fisher Vector representations, which are further fed into classifiers like

support vector machines (SVMs). In contrast to hand crafting features, which is usually time-consuming and requires domain knowledge, there is a recent trend to learn robust feature representations with deep learning from raw video data. In the following, we review two categories of deep learning algorithms for video classification, i.e., supervised deep learning and unsupervised feature learning.

1.3.1 Supervised Deep Learning for Classification

1.3.1.1 Image-Based Video Classification

The great success of CNN features on image analysis tasks [Girshick et al. 2014, Razavian et al. 2014] has stimulated the utilization of deep features for video classification. The general idea is to treat a video clip as a collection of frames, and then for each frame, feature representation could be derived by running a feed-forward pass till a certain fully-connected layer with state-of-the-art deep models pre-trained on ImageNet [Deng et al. 2009], including AlexNet [Krizhevsky et al. 2012], VGGNet [Simonyan and Zisserman 2015], GoogLeNet [Szegedy et al. 2015a], and ResNet [He et al. 2016b], as discussed earlier. Finally, frame-level features are averaged into video-level representations as inputs of standard classifiers for recognition, such as the well-known SVMs.

Among the works on image-based video classification, Zha et al. [2015] systematically studied the performance of image-based video recognition using features from different layers of deep models together with multiple kernels for classification. They demonstrated that off-the-shelf CNN features coupled with kernel SVMs can obtain decent recognition performance. Motivated by the advanced feature encoding strategies in images [Sánchez et al. 2013], Xu et al. [2015c] proposed to obtain video-level representation through vector of locally aggregated descriptors (VLAD) encoding [Jégou et al. 2010b], which can attain performance gain over the trivial averaging pooling approach. Most recently, Qiu et al. [2016] devised a novel Fisher Vector encoding with Variational AutoEncoder (FV-VAE) to quantize the local activations of the convolutional layer, which learns powerful visual representations of better generalization.

1.3.1.2 End-to-End CNN Architectures

The effectiveness of CNNs on a variety of tasks lies in their capability to learn features from raw data as an end-to-end pipeline targeting a particular task [Szegedy et al. 2015a, Long et al. 2015, Girshick 2015]. Therefore, in contrast to the image-based classification methods, there are many works focusing on applying CNN models to the video domain with an aim to learn hidden spatio-temporal patterns. Ji et al. [2010] introduced the 3D CNN model that operates on stacked video frames,

extending the traditional 2D CNN designed for images to the spatio-temporal space. The 3D CNN utilizes 3D kernels for convolution to learn motion information between adjacent frames in volumes segmented by human detectors. [Karpathy et al. \[2014\]](#) compared several similar architectures on a large scale video dataset in order to explore how to better extend the original CNN architectures to learn spatio-temporal clues in videos. They found that the performance of the CNN model with a single frame as input achieves similar results to models operating on a stack of frames, and they also suggested that a mixed-resolution architecture consisting of a low-resolution context and a high-resolution stream could speed up training effectively. Recently, [Tran et al. \[2015\]](#) also utilized 3D convolutions with modern deep architectures. However, they adopted full frames as the inputs of 3D CNNs instead of the segmented volumes in [Ji et al. \[2010\]](#).

Though the extension of conventional CNN models by stacking frames makes sense, the performance of such models is worse than that of state-of-the-art hand-crafted features [[Wang and Schmid 2013](#)]. This may be because the spatio-temporal patterns in videos are too complex to be captured by deep models with insufficient training data. In addition, the training of CNNs with inputs of 3D volumes is usually time-consuming. To effectively handle 3D signals, [Sun et al. \[2015\]](#) introduced factorized spatio-temporal convolutional networks that factorize the original 3D convolution kernel learning as a sequential process of learning 2D spatial kernels in the lower layer. In addition, motivated by the fact that videos can naturally be decomposed into spatial and temporal components, [Simonyan and Zisserman \[2014\]](#) proposed a two-stream approach (see [Figure 1.3](#)), which breaks down the learning of video representation into separate feature learning of spatial and temporal clues. More specifically, the authors first adopted a typical spatial CNN to model appearance information with raw RGB frames as inputs. To account for temporal clues among adjacent frames, they explicitly generated multiple-frame dense optical flow, upon which a temporal CNN is trained. The dense optical flow is derived from computing displacement vector fields between adjacent frames (see [Figure 1.4](#)), which represent motions in an explicit way, making the training of the network easier. Finally, at test time, each individual CNN generates a prediction by averaging scores from 25 uniformly sampled frames (optical flow frames) for a video clip, and then the final output is produced by the weighted sum of scores from the two streams. The authors reported promising results on two action recognition benchmarks. As the two-stream approach contains many implementation choices that may affect the performance, [Ye et al. \[2015b\]](#) evaluated different options, including dropout ratio and network architecture, and discussed their findings.

Very recently, there have been several extensions of the two-stream approach. Wang et al. utilized the point trajectories from the improved dense trajectories

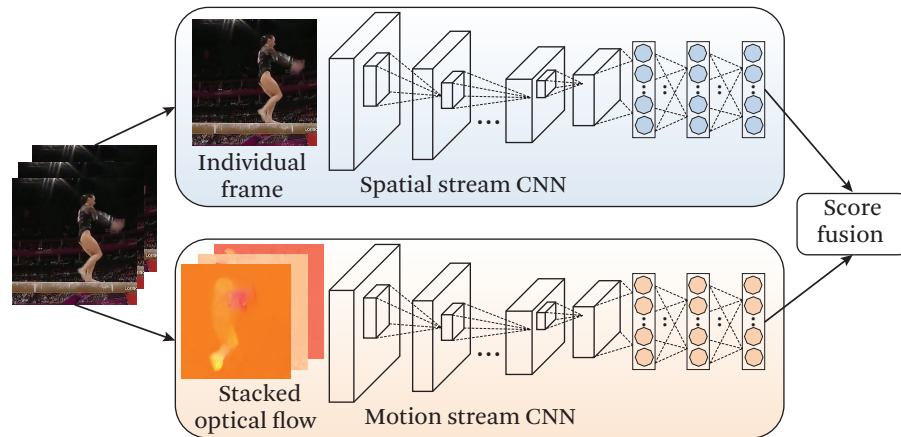


Figure 1.3 Two-stream CNN framework. (From [Wu et al. \[2015c\]](#))

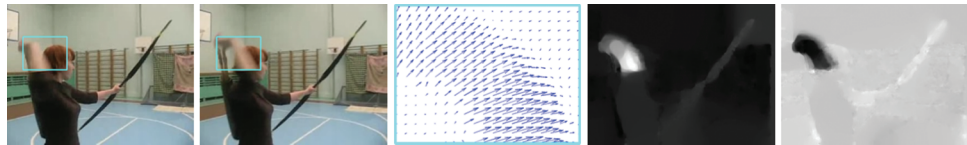


Figure 1.4 Examples of optical flow images. (From [Simonyan and Zisserman \[2014\]](#))

[[Wang and Schmid 2013](#)] to pool two-stream convolutional feature maps to generate trajectory-pooled deep-convolutional descriptors (TDD) [[Wang et al. 2015](#)]. [Feichtenhofer et al. \[2016\]](#) improved the two-stream approach by exploring a better fusion approach to combine spatial and temporal streams. They found that two streams could be fused using convolutional layers rather than averaging classification scores to better model the correlations of spatial and temporal streams. [Wang et al. \[2016b\]](#) introduced temporal segment networks, where each segment is used as the input of a two-stream network and the final prediction of a video clip is produced by a consensus function combining segment scores. [Zhang et al. \[2016\]](#) proposed to replace the optical flow images with motion vectors with an aim to achieve real-time action recognition. More recently, [Wang et al. \[2016c\]](#) proposed to learn feature representation by modeling an action as a transformation from an initial state (condition) to a new state (effect) with two Siamese CNN networks, operating on RGB frames and optical flow images. Similar to the original two-stream approach, they then fused the classification scores from two streams linearly to obtain final predictions. They reported better results on two challenging benchmarks

than [Simonyan and Zisserman \[2014\]](#), possibly because the transformation from precondition to effect could implicitly model the temporal coherence in videos. [Zhu et al. \[2016\]](#) proposed a key volume mining approach that attempts to identify key volumes and perform classification at the same time. [Bilen et al. \[2016\]](#) introduced the dynamic image to represent motions with rank pooling in videos, upon which a CNN model is trained for recognition.

1.3.1.3 Modeling Long-Term Temporal Dynamics

As discussed earlier, the temporal CNN in the two-stream approach [[Simonyan and Zisserman 2014](#)] explicitly captures the motion information among adjacent frames, which, however, only depicts movements within a short time window. In addition, during the training of CNN models, each sweep takes a single frame (or a stacked optical frame image) as the input of the network, failing to take the order of frames into account. This is not sufficient for video analysis, since complicated events/actions in videos usually consist of multiple actions happening over a long time. For instance, a “making pizza” event can be decomposed into several sequential actions, including “making the dough,” “topping,” and “baking.” Therefore, researchers have recently attempted to leverage RNN models to account for the temporal dynamics in videos, among which LSTM is a good fit without suffering from the “vanishing gradient” effect, and has demonstrated its effectiveness in several tasks like image/video captioning [[Donahue et al. 2017](#), [Yao et al. 2015a](#)] (to be discussed in detail later) and speech analysis [[Graves et al. 2013](#)].

[Donahue et al. \[2017\]](#) trained two two-layer LSTM networks (Figure 1.5) for action recognition with features from the two-stream approach. They also tried to fine-tune the CNN models together with LSTM but did not obtain significant performance gain compared with only training the LSTM model. [Wu et al. \[2015c\]](#) fused the outputs of LSTM models with CNN models to jointly model spatio-temporal clues for video classification and observed that CNNs and LSTMs are highly complementary. [Ng et al. \[2015\]](#) further trained a 5-layer LSTM model and compared several pooling strategies. Interestingly, the deep LSTM model performs on par with single frame CNN on a large YouTube video dataset called Sports-1M, which may be because the videos in this dataset are uploaded by ordinary users without professional editing and contain cluttered backgrounds and severe camera motion. [Veeriah et al. \[2015\]](#) introduced a differential gating scheme for LSTM to emphasize the change in information gain to remove redundancy in videos. Recently, in a multi-granular spatio-temporal architecture [[Li et al. 2016a](#)], LSTMs have been utilized to further model the temporal information of frame, motion, and clip streams. [Wu et al. \[2016\]](#) further employed a CNN operating on spectrograms derived from

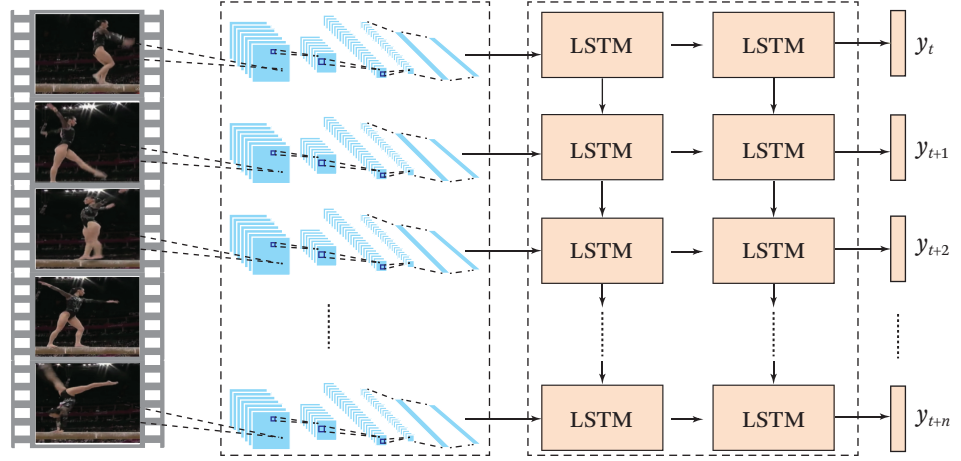


Figure 1.5 Utilizing LSTMs to explore temporal dynamics in videos with CNN features as inputs.

soundtracks of videos to complement visual clues captured by CNN and LSTMs, and demonstrated strong results.

1.3.1.4 Incorporating Visual Attention

Videos contain many frames. Using all of them is computationally expensive and may degrade the performance of recognizing a class of interest as not all the frames are relevant. This issue has motivated researchers to leverage the attention mechanism to identify the most discriminative spatio-temporal volumes that are directly related to the targeted semantic class. [Sharma et al. \[2015\]](#) proposed the first attention LSTM for action recognition with a soft-attention mechanism to attach higher importance to the learned relevant parts in video frames. More recently, [Li et al. \[2016c\]](#) introduced the VideoLSTM, which applied attention in convolutional LSTM models to discover relevant spatio-temporal volumes. In addition to soft-attention, VideoLSTM also employed motion-based attention derived from optical flow images for better action localization.

1.3.2 Unsupervised Video Feature Learning

Current remarkable improvements with deep learning heavily rely on a large amount of labeled data. However, scaling up to thousands of video categories presents significant challenges due to insurmountable annotation efforts even at video level, not to mention frame-level fine-grained labels. Therefore, the utilization of unsupervised learning, integrating spatial and temporal context information, is a promising way to find and represent structures in videos. [Taylor et al. \[2010\]](#)

proposed a convolutional gated boltzmann Machine to learn to represent optical flow and describe motion. [Le et al. \[2011\]](#) utilized two-layer independent subspace analysis (ISA) models to learn spatio-temporal models for action recognition. More recently, [Srivastava et al. \[2015\]](#) adopted an encoder-decoder LSTM to learn feature representations in an unsupervised way. They first mapped an input sequence into a fixed-length representation by an encoder LSTM, which would be further decoded with single or multiple decoder LSTMs to perform different tasks, such as reconstructing the input sequence, or predicting the future sequence. The model was first pre-trained on YouTube data without manual labels, and then fine-tuned on standard benchmarks to recognize actions. [Pan et al. \[2016a\]](#) explored both local temporal coherence and holistic graph structure preservation to learn a deep intrinsic video representation in an end-to-end fashion. [Ballas et al. \[2016\]](#) leveraged convolutional maps from different layers of a pre-trained CNN as the input of a gated recurrent unit (GRU)-RNN to learn video representations.

Summary

The latest developments discussed above have demonstrated the effectiveness of deep learning for video classification. However, current deep learning approaches for video classification usually resort to popular deep models in image and speech domain. The complicated nature of video data, containing abundant spatial, temporal, and acoustic clues, makes off-the-shelf deep models insufficient for video-related tasks. This highlights the need for a tailored network to effectively capture spatial and acoustic information, and most importantly to model temporal dynamics. In addition, training CNN/LSTM models requires manual labels that are usually expensive and time-consuming to acquire, and hence one promising direction is to make full utilization of the substantial amounts of unlabeled video data and rich contextual clues to derive better video representations.

1.4 Video Captioning

Video captioning is a new problem that has received increasing attention from both computer vision and natural language processing communities. Given an input video, the goal is to automatically generate a complete and natural sentence, which could have a great potential impact, for instance, on robotic vision or on helping visually impaired people. Nevertheless, this task is very challenging, as a description generation model should capture not only the objects, scenes, and activities presented in the video, but also be capable of expressing how these objects/scenes/activities relate to each other in a natural sentence. In this section, we elaborate the problem by surveying the state-of-the-art methods. We classify exist-

Input video:



Video tagging:
baby, boy, chair

Video (frame) captioning:
Two baby boys are in the chair.

Video captioning:
A baby boy is biting finger of another baby boy.

Figure 1.6 Examples of video tagging, image (frame) captioning, and video captioning. The input is a short video, while the output is a text response to this video, in the form of individual words (tags), a natural sentence describing one single image (frame), and dynamic video contents, respectively.

ing methods in terms of different strategies for sentence modeling. In particular, we distill a common architecture of combining convolutional and recurrent neural networks for video captioning. As video captioning is an emerging area, we start by introducing the problem in detail.

1.4.1 Problem Introduction

Although there has already been extensive research on video tagging [Siersdorfer et al. 2009, Yao et al. 2013] and image captioning [Vinyals et al. 2015, Donahue et al. 2017], video-level captioning has its own characteristics and thus is different from tagging and image/frame-level captioning. A video tag is usually the name of a specific object, action, or event, which is recognized in the video (e.g., “baby,” “boy,” and “chair” in Figure 1.6). Image (frame) captioning goes beyond tagging by describing an image (frame) with a natural sentence, where the spatial relationships between objects or object and action are further described (e.g., “Two baby boys are in the chair” generated on one single frame of Figure 1.6). Video captioning has been taken as an even more challenging problem, as a description should not only capture the above-mentioned semantic knowledge in the video but also express the spatio-temporal relationships in between and the dynamics in a natural sentence (e.g., “A baby boy is biting finger of another baby boy” for the video in Figure 1.6).

Despite the difficulty of the problem, there have been several attempts to address video caption generation [Pan et al. 2016b, Yu et al. 2016, Xu et al. 2016], which are mainly inspired by recent advances in machine translation [Sutskever

et al. 2014]. The elegant recipes behind this are the promising developments of the CNNs and the RNNs. In general, 2D [Simonyan and Zisserman 2015] and/or 3D CNNs [Tran et al. 2015] are exploited to extract deep visual representations and LSTM [Hochreiter and Schmidhuber 1997] is utilized to generate the sentence word by word. More sophisticated frameworks, additionally integrating internal or external knowledge in the form of high-level semantic attributes or further exploring the relationship between the semantics of sentence and video content, have also been studied for this problem.

In the following subsections we present a comprehensive review of video captioning methods through two main categories based on the strategies for sentence generation (Section 1.4.2) and generalizing a common architecture by leveraging sequence learning for video captioning (Section 1.4.3).

1.4.2 Approaches for Video Captioning

There are mainly two directions for video captioning: a template-based language model [Kojima et al. 2002, Rohrbach et al. 2013, Rohrbach et al. 2014, Guadarrama et al. 2013, Xu et al. 2015b] and sequence learning models (e.g., RNNs) [Donahue et al. 2017, Pan et al. 2016b, Xu et al. 2016, Yu et al. 2016, Venugopalan et al. 2015a, Yao et al. 2015a, Venugopalan et al. 2015b, Venugopalan et al. 2015b]. The former predefines the special rule for language grammar and splits the sentence into several parts (e.g., subject, verb, object). With such sentence fragments, many works align each part with detected words from visual content by object recognition and then generate a sentence with language constraints. The latter leverages sequence learning models to directly learn a translatable mapping between video content and sentence. We will review the state-of-the-art research along these two dimensions.

1.4.2.1 Template-based Language Model

Most of the approaches in this direction depend greatly on the sentence templates and always generate sentences with syntactical structure. Kojima et al. [2002] is one of the early works that built a concept hierarchy of actions for natural language description of human activities. Tan et al. [2011] proposed using predefined concepts and sentence templates for video event recounting. Rohrbach et al.'s conditional random field (CRF) learned to model the relationships between different components of the input video and generate descriptions for videos [Rohrbach et al. 2013]. Furthermore, by incorporating semantic unaries and hand-centric features, Rohrbach et al. [2014] utilized a CRF-based approach to generate coherent video descriptions. In 2013, Guadarrama et al. used semantic hierarchies to choose an appropriate level of the specificity and accuracy of sentence fragments. Recently, a

deep joint video-language embedding model in [Xu et al. \[2015b\]](#) was designed for video sentence generation.

1.4.2.2 Sequence Learning

Unlike the template-based language model, sequence learning-based methods can learn the probability distribution in the common space of visual content and textual sentence and generate novel sentences with more flexible syntactical structure. [Donahue et al. \[2017\]](#) employed a CRF to predict activity, object, and location present in the video input. These representations were concatenated into an input sequence and then translated to a natural sentence with an LSTM model. Later, [Venugopalan et al. \[2015b\]](#) proposed an end-to-end neural network to generate video descriptions by reading only the sequence of video frames. By mean pooling, the features over all the frames can be represented by one single vector, which is the input of the following LSTM model for sentence generation. [Venugopalan et al. \[2015a\]](#) then extended the framework by inputting both frames and optical flow images into an encoder-decoder LSTM. Inspired by the idea of learning visual-semantic embedding space in search [[Pan et al. 2014](#), [Yao et al. 2015b](#)], [[Pan et al. 2016b](#)] additionally considered the relevance between sentence semantics and video content as a regularizer in LSTM based architecture. In contrast to mean pooling, [Yao et al. \[2015a\]](#) proposed to utilize the temporal attention mechanism to exploit temporal structure as well as a spatio-temporal convolutional neural network to obtain local action features. Then, the resulting video representations were fed into the text-generating RNN. In addition, similar to the knowledge transfer from image domain to video domain [[Yao et al. 2012, 2015c](#)], [Liu and Shi \[2016\]](#) leveraged the learned models on image captioning to generate a caption for each video frame and incorporate the obtained captions, regarded as the attributes of each frame, into a sequence-to-sequence architecture to generate video descriptions. Most recently, with the encouraging performance boost reported on the image captioning task by additionally utilizing high-level image attributes in [Yao et al. \[2016\]](#), [Pan et al. \[2016c\]](#) further leveraged semantic attributes learned from both images and videos with a transfer unit for enhancing video sentence generation.

1.4.3 A Common Architecture for Video Captioning

To better summarize the frameworks of video captioning by sequence learning, we illustrate a common architecture as shown in [Figure 1.7](#). Given a video, 2D and/or 3D CNNs are utilized to extract visual features on raw video frames, optical flow images, and video clips. The video-level representations are produced by mean

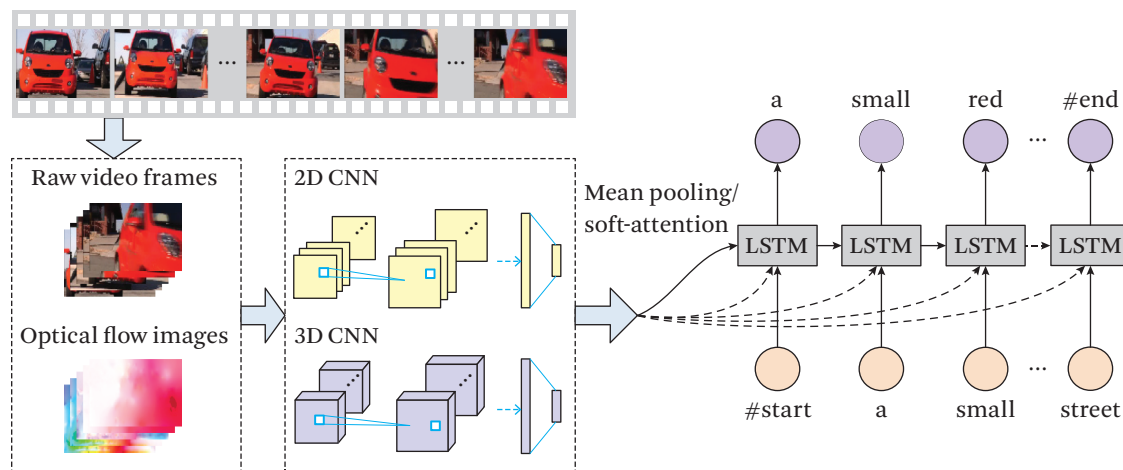


Figure 1.7 A common architecture for video captioning by sequence learning. The video representations are produced by mean pooling or soft-attention over the visual features of raw video frames/optical flow images/video clips, extracted by 2D/3D CNNs. The sentence is generated word by word in the following LSTM, based on the video representations.

pooling or soft attention over these visual features. Then, an LSTM is trained for generating a sentence based on the video-level representations.

Technically, suppose we have a video \mathcal{V} with N_v sample frames/optical images/clips (uniform sampling) to be described by a textual sentence \mathcal{S} , where $\mathcal{S} = \{w_1, w_2, \dots, w_{N_s}\}$ consisting of N_s words. Let $\mathbf{v} \in \mathbb{R}^{D_v}$ and $\mathbf{w}_t \in \mathbb{R}^{D_w}$ denote the D_v -dimensional visual features of a video \mathcal{V} and the D_w -dimensional textual features of the t -th word in sentence \mathcal{S} , respectively. As a sentence consists of a sequence of words, a sentence can be represented by a $D_w \times N_s$ matrix $\mathbf{W} \equiv [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{N_s}]$, with each word in the sentence as its column vector. Hence, given the video representations \mathbf{v} , we aim to estimate the conditional probability of the output word sequence $\{w_1, w_2, \dots, w_{N_s}\}$, i.e.,

$$\Pr(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{N_s} | \mathbf{v}). \quad (1.3)$$

Since the model produces one word in the sentence at each time step, it is natural to apply the chain rule to model the joint probability over the sequential words. Thus, the log probability of the sentence is given by the sum of the log probabilities over the words and can be expressed as:

$$\log \Pr(\mathbf{W} | \mathbf{v}) = \sum_{t=1}^{N_s} \log \Pr(\mathbf{w}_t | \mathbf{v}, \mathbf{w}_1, \dots, \mathbf{w}_{t-1}). \quad (1.4)$$

In the model training, we feed the start sign word *#start* into LSTM, which indicates the start of the sentence generation process. We aim to maximize the log probability of the output video description \mathcal{S} given the video representations, the previous words it has seen, and the model parameters θ , which can be formulated as

$$\theta^* = \arg \max_{\theta} \sum_{t=1}^{N_s} \log \Pr(\mathbf{w}_t | \mathbf{v}, \mathbf{w}_1, \dots, \mathbf{w}_{t-1}; \theta). \quad (1.5)$$

This log probability is calculated and optimized over the whole training dataset using stochastic gradient descent. Note that the end sign word *#end* is required to terminate the description generation. During inference, we choose the word with maximum probability at each time step and set it as the LSTM input for the next time step until the end sign word is emitted.

Summary

The introduction of the video captioning problem is relatively new. Recently, this task has sparked significant interest and may be regarded as the ultimate goal of video understanding. Video captioning is a complex problem and has been initially forwarded by the fundamental technological advances in recognition that can effectively recognize key objects or scenes from video contents. The developments of RNNs in machine translation have further accelerated the growth of this research direction. The recent results, although encouraging, are still indisputably far from practical use, as the forms of the generated sentences are simple and the vocabulary is still limited. How to generate free-form sentences and support open vocabulary are vital issues for the future of this task.

1.5 Benchmarks and Challenges

We now discuss popular benchmarks and challenges for video classification (Section 1.5.1) and video captioning (Section 1.5.2).

1.5.1 Classification

Research on video classification has been stimulated largely by the release of the large and challenging video datasets such as UCF101 [Soomro et al. 2012], HMDB51 [Kuehne et al. 2011], and FCVID [Jiang et al. 2015], and by the open challenges organized by fellow researchers, including the THUMOS challenge [Jiang et al. 2014b], the ActivityNet Large Scale Activity Recognition Challenge [Heilbron et al. 2015], and the TRECVID multimedia event detection (MED) task [Over et al. 2014].

Table 1.1 Popular benchmark datasets for video classification, sorted by the year of construction

Dataset	#Video	#Class	Released Year	Background
KTH	600	6	2004	Clean Static
Weizmann	81	9	2005	Clean Static
Kodak	1,358	25	2007	Dynamic
Hollywood	430	8	2008	Dynamic
Hollywood2	1,787	12	2009	Dynamic
MCG-WEBV	234,414	15	2009	Dynamic
Olympic Sports	800	16	2010	Dynamic
HMDB51	6,766	51	2011	Dynamic
CCV	9,317	20	2011	Dynamic
UCF-101	13,320	101	2012	Dynamic
THUMOS-2014	18,394	101	2014	Dynamic
MED-2014 (Dev. set)	≈31,000	20	2014	Dynamic
Sports-1M	1,133,158	487	2014	Dynamic
ActivityNet	27,901	203	2015	Dynamic
EventNet	95,321	500	2015	Dynamic
MPII Human Pose	20,943	410	2014	Dynamic
FCVID	91,223	239	2015	Dynamic

In the following, we first discuss related datasets according to the list shown in Table 1.1, and then summarize the results of existing works.

1.5.1.1 Datasets

KTH dataset is one of the earliest benchmarks for human action recognition [Schuldt et al. 2004]. It contains 600 short videos of 6 human actions performed by 25 people in four different scenarios.

Weizmann dataset is another very early and simple dataset, consisting of 81 short videos associated with 9 actions performed by 9 actors [Blank et al. 2005].

Kodak Consumer Videos dataset was recorded by around 100 customers of the Eastman Kodak Company [Loui et al. 2007]. The dataset collected 1,358

video clips labeled with 25 concepts (including activities, scenes, and single objects) as a part of the Kodak concept ontology.

Hollywood Human Action dataset contains 8 action classes collected from 32 Hollywood movies, totaling 430 video clips [Laptev et al. 2008]. It was further extended to the Hollywood2 [Marszalek et al. 2009] dataset, which is composed of 12 actions from 69 Hollywood movies with 1,707 video clips in total. This Hollywood series is challenging due to cluttered background and severe camera motion throughout the datasets.

MCG-WEBV dataset is another large set of YouTube videos that has 234,414 web videos with annotations on several topic-level events like “a conflict at Gaza” [Cao et al. 2009].

Olympic Sports includes 800 video clips and 16 action classes [Niebles et al. 2010]. It was first introduced in 2010 and, unlike in previous datasets, all the videos were downloaded from the Internet.

HMDB51 dataset comprises 6,766 videos annotated into 51 classes [Kuehne et al. 2011]. The videos are from a variety of sources, including movies and YouTube consumer videos.

Columbia Consumer Videos (CCV) dataset was constructed in 2011, aiming to stimulate research on Internet consumer video analysis [Jiang et al. 2011]. It contains 9,317 user-generated videos from YouTube, which were annotated into 20 classes, including objects (e.g., “cat” and “dog”), scenes (e.g., “beach” and “playground”), sports events (e.g., “basketball” and “soccer”), and social activities (e.g., “birthday” and “graduation”).

UCF-101 & THUMOS-2014 dataset is another popular benchmark for human action recognition in videos, consisting of 13,320 video clips (27 hours in total) with 101 annotated classes such as “diving” and “weight lifting” [Soomro et al. 2012]. More recently, the THUMOS-2014 Action Recognition Challenge [Jiang et al. 2014b] created a benchmark by extending the UCF-101 dataset (used as the training set). Additional videos were collected from the Internet, including 2,500 background videos, 1,000 validation videos, and 1,574 test videos.

TRECVID MED dataset was released and annually updated by the task of MED, created by NIST since 2010 [Over et al. 2014]. Each year an extended dataset based on datasets from challenges of previous years is constructed and released for worldwide system comparison. For example, in 2014 the MED

dataset contained 20 events, such as “birthday party,” “bike trick,” etc. According to NIST, in the development set, there are around 8,000 videos for training and 23,000 videos used as dry-run validation samples (1,200 hours in total). The MED dataset is only available to the participants of the task, and the labels of the official test set (200,000 videos) are not available even to the participants.

Sports-1M dataset consists of 1 million YouTube videos in 487 classes, such as “bowling,” “cycling,” “rafting,” etc., and has been available since 2014 [Karpathy et al. 2014]. The video annotations were automatically derived by analyzing online textual contexts of the videos. Therefore the labels of this dataset are not clean, but the authors claim that the quality of annotation is fairly good.

ActivityNet dataset is another large-scale video dataset for human activity recognition and understanding and was released in 2015 [Heilbron et al. 2015]. It consists of 27,801 video clips annotated into 203 activity classes, totaling 849 hours of video. Compared with existing datasets, ActivityNet contains more fine-grained action categories (e.g., “drinking beer” and “drinking coffee”).

EventNet dataset consists of 500 events and 4,490 event-specific concepts and was released in 2015 [Ye et al. 2015a]. It includes automatic detection models for its video events and some constituent concepts, with around 95,000 training videos from YouTube. Similarly to Sports-1M, EventNet was labeled by online textual information rather than manually labeled.

MPII Human Pose dataset includes around 25,000 images containing over 40,000 people with annotated body joints [Andriluka et al. 2014]. According to an established taxonomy of human activities (410 in total), the collected images (from YouTube videos) were provided with activity labels.

Fudan-Columbia Video Dataset (FCVID) dataset contains 91,223 web videos annotated manually into 239 categories [Jiang et al. 2015]. The categories cover a wide range of topics, such as social events (e.g., “tailgate party”), procedural events (e.g., “making cake”), object appearances (e.g., “panda”), and scenes (e.g., “beach”).

1.5.1.2 Challenges

To advance the state of the art in video classification, several challenges have been introduced with the aim of exploring and evaluating new approaches in realistic settings. We briefly introduce three representative challenges here.

THUMOS Challenge was first introduced in 2013 in the computer vision community, aiming to explore and evaluate new approaches for large-scale action recognition of Internet videos [Idrees et al. 2016]. The three editions of the challenge organized in 2013–2015 made THUMOS a common benchmark for action classification and detection.

TRECVID Multimedia Event Detection (MED) Task aims to detect whether a video clip contains an instance of a specific event [Awad et al. 2016, Over et al. 2015, Over et al. 2014]. Specifically, based on the released TRECVID MED dataset each year, each participant is required to provide for each testing video the confidence score of how likely one particular event is to happen in the video. Twenty pre-specified events are used each year, and this task adopts the metrics of average precision (AP) and inferred AP for event detection. Each event was also complemented with an event kit, i.e., the textual description of the event as well as the potentially useful information about related concepts that are likely contained in the event.

ActivityNet Large Scale Activity Recognition Challenge was first organized as a workshop in 2016 [Heilbron et al. 2015]. This challenge is based on the ActivityNet dataset [Heilbron et al. 2015], with the aim of recognizing high-level and goal-oriented activities. By using 203 activity categories, there are two tasks in this challenge: (1) Untrimmed Classification Challenge, and (2) Detection Challenge, which is to predict the labels and temporal extents of the activities present in videos.

1.5.1.3 Results of Existing Methods

Some of the datasets introduced above have been popularly adopted in the literature. We summarize the results of several recent approaches on UCF-101 and HMDB51 in Table 1.2, where we can see the fast pace of development in this area. Results on video classification are mostly measured by the AP (for a single class) and mean AP (for multiple classes), which are not introduced in detail as they are well known.

1.5.2 Captioning

A number of datasets have been proposed for video captioning; these commonly contain videos that have each been paired with its corresponding sentences annotated by humans. This section summarizes the existing datasets and the adopted evaluation metrics, followed by quantitative results of representative methods.

Table 1.2 Comparison of recent video classification methods on UCF-101 and HMDB51 datasets

Methods	UCF-101	HMDB51
LRCN [Donahue et al. 2017]	82.9	—
LSTM-composite [Srivastava et al. 2015]	84.3	—
F_{STCN} [Sun et al. 2015]	88.1	59.1
C3D [Tran et al. 2015]	86.7	—
Two-Stream [Simonyan and Zisserman 2014]	88.0	59.4
LSTM [Ng et al. 2015]	88.6	—
Image-Based [Zha et al. 2015]	89.6	—
Transformation CNN [Wang et al. 2016c]	92.4	63.4
Multi-Stream [Wu et al. 2016]	92.6	—
Key Volume Mining [Zhu et al. 2016]	92.7	67.2
Convolutional Two-Stream [Feichtenhofer et al. 2016]	93.5	69.2
Temporal Segment Networks [Wang et al. 2016b]	94.2	69.4

Table 1.3 Comparison of video captioning benchmarks

Dataset	Context	Sentence Source	#Videos	#Clips	#Sentences	#Words
MSVD	Multi-category	AMT workers	—	1,970	70,028	607,339
TV16-VTT	Multi-category	Humans	2,000	—	4,000	—
YouCook	Cooking	AMT workers	88	—	2,668	42,457
TACoS-ML	Cooking	AMT workers	273	14,105	52,593	—
M-VAD	Movie	DVS	92	48,986	55,905	519,933
MPII-MD	Movie	Script+DVS	94	68,337	68,375	653,467
MSR-VTT-10K	20 categories	AMT workers	7,180	10,000	200,000	1,856,523

1.5.2.1 Datasets

Table 1.3 summarizes key statistics and comparisons of popular datasets for video captioning. Figure 1.8 shows a few examples from some of the datasets.

Microsoft Research Video Description Corpus (MSVD) contains 1,970 YouTube snippets collected on Amazon Mechanical Turk (AMT) by requesting workers to pick short clips depicting a single activity [Chen and Dolan 2011]. Annotators then label the video clips with single-sentence descrip-

(a) MSVD dataset**Sentences:**

- A dog walks around on its front legs.
- The dog is doing a handstand.
- A pug is trying for balance walk on two legs.

**Sentences:**

- A man lights a match book on fire.
- A man playing with fire sticks.
- A man lights matches and yells.

(b) M-VAD dataset**Sentence:**

- Later he drags someone through a jog.

**Sentence:**

- A waiter brings a pastry with a candle.

(c) MPII-MD dataset**Sentence:**

- He places his hands around her waist as she opens her eyes.

**Sentence:**

- Someone's car is stopped by a couple of uniformed police.

(d) MSR-VTT-10K dataset**Sentences:**

- People practising volleyball in the play ground.
- A man is hitting a ball and he falls.
- A man is playing a football game on green land.

**Sentences:**

- A cat is hanging out in a bassinet with a baby.
- The cat is in the baby bed with the baby.
- A cat plays with a child in a crib.

Figure 1.8 Examples from (a) MSVD, (b) M-VAD, (c) MPII-MD, and (d) MSR-VTT-10K datasets.

tions. The original corpus has multi-lingual descriptions, but only the English descriptions are commonly exploited on video captioning tasks. Specifically, there are roughly 40 available English descriptions per video and the standard split of MSVD is 1,200 videos for training, 100 for validation, and 670 for testing, as suggested in [Guadarrama et al. \[2013\]](#).

YouCook dataset consists of 88 in-house cooking videos crawled from YouTube and is roughly uniformly split into six different cooking styles, such as baking and grilling [[Das et al. 2013](#)]. All the videos are in a third-person viewpoint and in different kitchen environments. Each video is annotated with multiple human descriptions by AMT. Each annotator in AMT is instructed to describe the video in at least three sentences totaling a minimum of 15 words, resulting in 2,668 sentences for all the videos.

TACoS Multi-Level Corpus (TACoS-ML) is mainly built [[Rohrbach et al. 2014](#)] based on MPII Cooking Activities dataset 2.0 [[Rohrbach et al. 2015c](#)], which records different activities used when cooking. TACoS-ML consists of 185 long videos with text descriptions collected via AMT workers. Each AMT worker annotates a sequence of temporal intervals across the long video, pairing every interval with a single short sentence. There are 14,105 distinct intervals and 52,593 sentences in total.

Montreal Video Annotation Dataset (M-VAD) is composed of about 49,000 DVD movie snippets, which are extracted from 92 DVD movies [[Torabi et al. 2015](#)]. Each movie clip is accompanied by one single sentence from semi-automatically transcribed descriptive video service (DVS) narrations. The fact that movies always contain a high diversity of visual and textual content, and that there is only one single reference sentence for each movie clip, has made the video captioning task on the M-VAD dataset very challenging.

MPII Movie Description Corpus (MPII-MD) is another collection of movie descriptions dataset that is similar to M-VAD [[Rohrbach et al. 2015b](#)]. It contains around 68,000 movie snippets from 94 Hollywood movies and each snippet is labeled with one single sentence from movie scripts and DVS.

MSR Video to Text (MSR-VTT-10K) is a recent large-scale benchmark for video captioning that contains 10K Web video clips totalling 41.2 hours, covering the most comprehensive 20 categories obtained from a commercial video

search engine, e.g., music, people, gaming, sports, and TV shows [Xu et al. 2016]. Each clip is annotated with about 20 natural sentences by AMT workers. The training/validation/test split is provided by the authors with 6,513 clips for training, 2,990 for validation, and 497 for testing.

The TRECVID 2016 Video to Text Description (TV16-VTT) is another recent video captioning dataset that consists of 2,000 videos randomly selected from Twitter Vine videos [Awad et al. 2016]. Each video has a total duration of about 6 seconds and is annotated with 2 sentences by humans. The human annotators are asked to address four facets in the generated sentences: who the video is describing (kinds of persons, animals, things) and what the objects and beings are doing, plus where it is taking place and when.

1.5.2.2 Evaluation Metrics

For quantitative evaluation of the video captioning task, three metrics are commonly adopted: BLEU@ N [Papineni et al. 2002], METEOR [Banerjee and Lavie 2005], and CIDEr [Vedantam et al. 2015]. Specifically, BLEU@ N is a popular machine translation metric which measures the fraction of N -gram (up to 4-gram) in common between a hypothesis and a reference or set of references. However, as pointed out in Chen et al. [2015], the N -gram matches for a high N (e.g., 4) rarely occur at a sentence level, resulting in poor performance of BLEU@ N especially when comparing individual sentences. Hence, a more effective evaluation metric, METEOR, utilized along with BLEU@ N , is also widely used in natural language processing (NLP) community. Unlike BLEU@ N , METEOR computes uni-gram precision and recall, extending exact word matches to include similar words based on WordNet synonyms and stemmed tokens. Another important metric for image/video captioning is CIDEr, which measures consensus in image/video captioning by performing a Term Frequency Inverse Document Frequency (TF-IDF) weighting for each N -gram.

1.5.2.3 Results of Existing Methods

Most popular methods of video captioning have been evaluated on MSVD [Chen and Dolan 2011], M-VAD [Torabi et al. 2015], MPII-MD [Rohrbach et al. 2015b], and TACoS-ML [Rohrbach et al. 2014] datasets. We summarize the results on these four datasets in Tables 1.4, 1.5, and 1.6. As can be seen, most of the works are very recent, indicating that video captioning is an emerging and fast-developing research topic.

Table 1.4 Reported results on the MSVD dataset, where $B@N$, M, and C are short for $BLEU@N$, METEOR, and CIDEr-D scores, respectively

Methods	B@1	B@2	B@3	B@4	M	C
FGM [Thomason et al. 2014]	—	—	—	13.7	23.9	—
LSTM-YT [Venugopalan et al. 2015b]	—	—	—	33.3	29.1	—
MM-VDN [Xu et al. 2015a]	—	—	—	37.6	29.0	—
S2VT [Venugopalan et al. 2015a]	—	—	—	—	29.8	—
S2FT [Liu and Shi 2016]	—	—	—	—	29.9	—
SA [Yao et al. 2015a]	80.0	64.7	52.6	41.9	29.6	51.7
Glove+Deep Fusion [Venugopalan et al. 2016]	—	—	—	42.1	31.4	—
LSTM-E [Pan et al. 2016b]	78.8	66.0	55.4	45.3	31.0	—
GRU-RCN [Ballas et al. 2016]	—	—	—	43.3	31.6	68.0
h-RNN [Yu et al. 2016]	81.5	70.4	60.4	49.9	32.6	65.8

All values are reported as percentages (%).

Table 1.5 Reported results on (a) M-VAD and (b) MPII-MD datasets, where M is short for METEOR

M-VAD dataset	
Methods	M
SA [Yao et al. 2015a]	4.3
Mean Pool [Venugopalan et al. 2015a]	6.1
Visual-Labels [Rohrbach et al. 2015a]	6.4
S2VT [Venugopalan et al. 2015a]	6.7
Glove+Deep Fusion [Venugopalan et al. 2016]	6.8
LSTM-E [Pan et al. 2016b]	6.7
MPII-MD dataset	
Methods	M
SMT [Rohrbach et al. 2015b]	5.6
Mean Pool [Venugopalan et al. 2015a]	6.7
Visual-Labels [Rohrbach et al. 2015a]	7.0
S2VT [Venugopalan et al. 2015a]	7.1
Glove+Deep Fusion [Venugopalan et al. 2016]	6.8
LSTM-E [Pan et al. 2016b]	7.3

All values are reported as percentages (%).

Table 1.6 Reported results on the TACoS-ML dataset, where $B@N$, M , and C are short for $BLEU@N$, METEOR, and CIDEr-D scores, respectively

Methods	$B@1$	$B@2$	$B@3$	$B@4$	M	C
CRF-T [Rohrbach et al. 2013]	56.4	44.7	33.2	25.3	26.0	124.8
CRF-M [Rohrbach et al. 2014]	58.4	46.7	35.2	27.3	27.2	134.7
LRCN [Donahue et al. 2017]	59.3	48.2	37.0	29.2	28.2	153.4
h-RNN [Yu et al. 2016]	60.8	49.6	38.5	30.5	28.7	160.2

All values are reported as percentages (%).

1.6 Conclusion

In this chapter, we have reviewed state-of-the-art deep learning techniques on two key topics related to video analysis, video classification and video captioning, both of which rely on the modeling of the abundant spatial and temporal information in videos. In contrast to hand crafted features that are costly to design and have limited generalization capability, the essence of deep learning for video classification is to derive robust and discriminative feature representations from raw data through exploiting massive videos with an aim to achieve effective and efficient recognition, which could hence serve as a fundamental component in video captioning. Video captioning, on the other hand, focuses on bridging visual understanding and language description by joint modeling. We also provided a review of popular benchmarks and challenges for both video classification and captioning tasks. Though extensive efforts have been made in video classification and captioning with deep learning, we believe we are just beginning to unleash the power of deep learning in the big video data era. Given the substantial amounts of videos generated at an astounding speed every hour and every day, it remains a challenging open problem how to derive better video representations with deep learning modeling the abundant interactions of objects and their evolution over time with limited (or without any) supervisory signals to facilitate video content understanding (i.e., the recognition of human activities and events as well as the generation of free-form and open-vocabulary sentences for describing videos). We hope this chapter sheds light on the nuts and bolts of video classification and captioning for both current and new researchers.



Audition for Multimedia Computing

Gerald Friedland (University of California, Berkeley),

Paris Smaragdis (University of Illinois

at Urbana-Champaign),

Josh McDermott (Massachusetts Institute of Technology),

Bhiksha Raj (Carnegie Mellon University)

What do the fields of robotics, human–computer interaction, AI, video retrieval, privacy, cybersecurity, Internet of Things, and big data all have in common? They all work with various sources of data: visual, textual, time stamps, links, records. But there is one source of data that has been almost completely ignored by the academic community—sound.

Our comprehension of the world relies critically on audition—the ability to perceive and interpret the sounds we hear. Sound is ubiquitous, and is a unique source of information about our environment and the events occurring in it. Just by listening, we can determine whether our child’s laughter originated inside or outside our house, how far away they were when they laughed, and whether the window through which the sound passed was open or shut. The ability to derive information about the world from sound is a core aspect of perceptual intelligence.

Auditory inferences are often complex and sophisticated despite their routine occurrence. The number of possible inferences is typically not enumerable, and the final interpretation is not merely one of selection from a fixed set. And yet humans perform such inferences effortlessly, based only on sounds captured using two sensors, our ears.

Electronic devices can also “perceive” sound. Every phone and tablet has at least one microphone, as do most cameras. Any device or space can be equipped

with microphones at minimal expense. Indeed, machines can not only “listen”; they have potential advantages over humans as listening devices, in that they can communicate and coordinate their experiences in ways that biological systems simply cannot. Collections of devices that can sense sound and communicate with each other could instantiate a single electronic entity that far surpasses humans in its ability to record and process information from sound.

And yet machines at present cannot truly hear. Apart from well-developed efforts to recover structure in speech and music, the state of the art in machine hearing is limited to relatively impoverished descriptions of recorded sounds: detecting occurrences of a limited pre-specified set of sound types, and their locations. Although researchers typically envision artificially intelligent agents such as robots to have human-like hearing abilities, at present the rich descriptions and inferences humans can make about sound are entirely beyond the capability of machine systems.

In this chapter, we suggest establishing the field of Computer Audition to develop the theory behind artificial systems that extract information from sound. Our objective is to enable computer systems to replicate and exceed human abilities. This chapter describes the challenges of this field.

2.1 A New Field

As just stated, the goal for the field of Computer Audition is to enable machine systems to replicate and surpass the inferences humans make from sound. The central challenge is that human listening abilities extend far beyond the simple localization and limited classification that are the mainstay of contemporary machine perception. The rich inferences we can make from sound enable us to answer a vast range of questions about what caused the sound. We consider this ability to be a defining characteristic of auditory perceptual intelligence, and seek machine systems with similar sophistication. Not only are we not restricted to fixed vocabularies of sounds, but we would be able to identify structure in sounds we’ve never heard before (as when we repeat back unfamiliar words in a spoken sentence), infer and account for interactions of different effects (e.g., the effects of distance and source intensity), and make inferences about causes (as when a jar falls on the floor).

In Computer Audition, we define “acoustic intelligence” as the ability to make inferences about the world and answer questions based on analysis of sound. This represents a basic change of perspective from the usual approach, which attempts

to *describe* what was “heard.” Instead, we suggest approaching the problem as that of deriving inferences from sound to *respond* to posed queries.

The primary distinction between the two approaches is that the former attempts to explicitly describe audio in terms of a well-defined set of sound events or phenomena, e.g., “engine sound,” “baby crying,” etc. The expressiveness of such analysis is limited to the size of the set of sound events/phenomena for which we possess models; the expressiveness can only be enhanced by increasing the size of this set. To use a Boolean analogy, it is based on explicit evaluation of an enumerable set of propositions; propositions outside this set cannot be verified.

The alternative approach that we advocate is, by contrast, open ended. An “acoustically intelligent” system must be able to respond to *any* sound-related query; the inferences derived from the sound must now be query-specific. Queries could, in principle, be unrestricted, ranging from the simple descriptors of the usual classification-based analysis(e.g., “Did we hear an ambulance?”) to those needing higher-level inferences—for example, *instance-level inferences* (“Did we just hear an accident?”), *temporal inferences* (“Did X happen before Y”), *patterns* (“Does X *generally* happen before Y when Z?”), *meta inferences* (“Was the window closed?” or “Did my child just trip?”), *spatial inferences* (“Where is X in relation to Y?”), *structural questions* (“Have we heard something like this before?”), *semantic inferences* (“Is this an abnormal heartbeat?”) or (“Is this an abnormal heartbeat pattern that we have in our archives?”), and so on.

In order to realize such intelligence, many challenges must be addressed, some of which are illustrated in Figure 2.1. It is important to note that our goal is to enable an open audio intelligence that will allow users to compose novel queries, and not just a system that only operates with prepackaged questions. We expand on some of the challenges in this chapter.

We believe Computer Audition systems will need to infer structured representations of generative processes in the world rather than simply learning to relate class labels to input signals. These challenges require an academic field that is willing to take risks that are not in the short-term interest of the computing industry. This field will be uniquely positioned inside multimedia computing, next to similar fields such as Computer Vision and Natural Language Processing (see Figure 2.2).

We position this new field of Computer Audition clearly inside multimedia computing. Not only are there analogies to Computer Vision and other multimedia-related fields but, historically, audio analysis research has been performed on corpora that were designed for a specific task, such as speech recognition, speaker identification, language recognition, etc. Having been constructed to serve specific

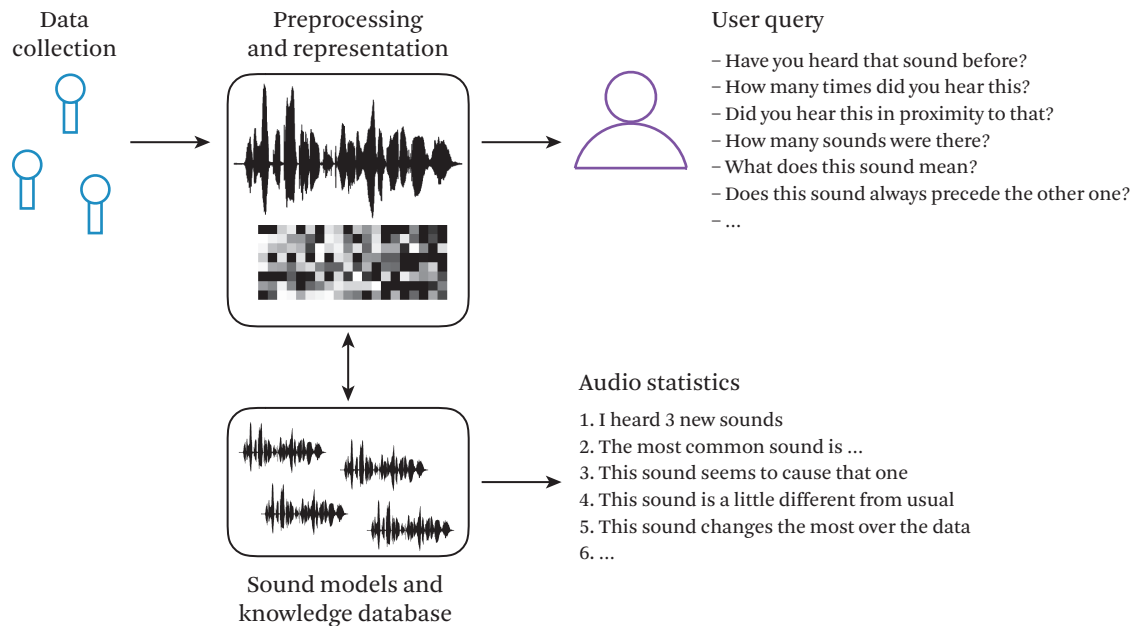


Figure 2.1 An exemplary computational audition system. Data collection is followed by transformation to more informative representations. Cross-referencing with multiple pre-trained sound models will result in initial semantic information that can be leveraged to create a list of automatically created observations and statistics, as well as to facilitate multiple user queries. These systems should be designed to be flexible, i.e., applicable to different audio domains—e.g., bio-acoustical data, music recordings, mechanical vibration readings, ecosystem recordings, etc.—and will allow for queries from domain experts who want to extract useful information.

research questions, these corpora are not representative of the diversity of signals occurring on the Internet, or by extension, in the natural world itself.

The popularity of published consumer-produced videos, however, makes wild audio data available at never-before-seen scale and therefore, for the first time in history, has made it possible to perform generic audio research. Handling generic audio with varied background noises, building classifiers for a large set of events, or coping with overlapping sounds are some of the problems that have been neglected due to the historical focus on carefully curated speech and music processing corpora. Moreover, work on consumer-produced videos lends itself better to exploring multimodal solutions.

Distinct from most existing work in audio recognition, the field should be principally oriented toward the understanding of “natural audio” (i.e., audio not based

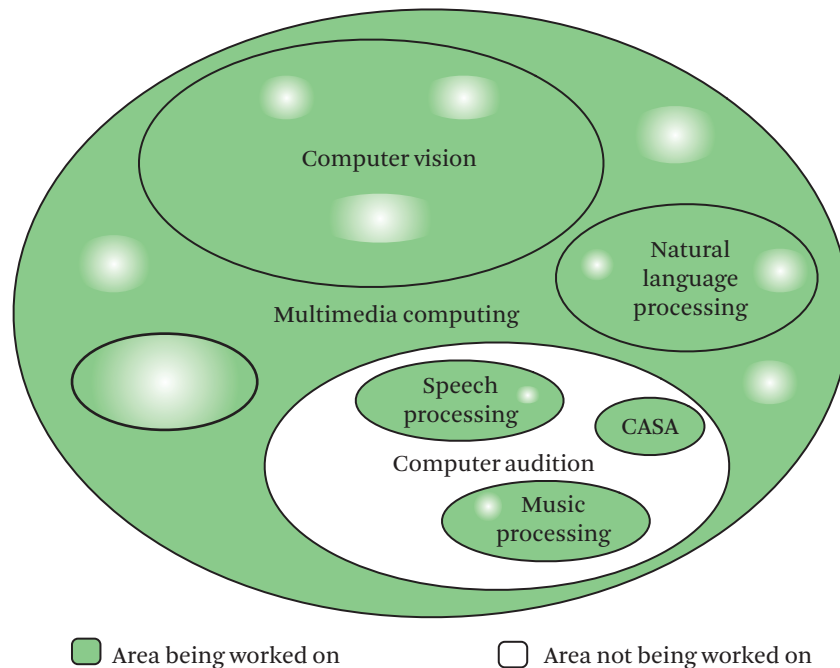


Figure 2.2 Positioning Computer Audition as a field relative to related disciplines; the white region has been neglected, in part due to lack of relevant data.

on lab-produced corpora, but recorded in natural environments), and will investigate automatic approaches that allow computers to “listen” and make sensible analyses of the components of natural audio. The field will encompass a range of areas, including signal processing, to derive representations that enable analysis of complex audio signals, pattern modeling, and recognition for detection of concepts and their relationship to one another, assignment of semantics to sound with and without labeled examples, and inference algorithms to analyze complex mixtures. Together they will enable an automated decomposition of audio to provide various forms of description that effectively represent a comprehension of the audio, and can facilitate the human-like analyses illustrated above.

2.2 Background

The topic of general understanding of natural audio for purposes of Computer Audition has not yet seen much attention in the multimedia nor the general academic community, as most of the research in the area was based on corpus-data created in

laboratories. A subset of previous work for natural audio analysis and understanding include, acoustic event detection, overlap detection and separation of different acoustic sources, blind source separation, computational auditory scene analysis, music signal processing, user verification, location estimation, the automatic discovery of atomic acoustic units, and various speech-related tasks (such as speech recognition, speaker recognition, and diarization). Past work in audio analysis and classification that is relevant to Computer Audition falls into three broad areas: speech recognition, source separation, and environmental sound recognition.

Speech recognition is the most intensely studied topic in audio, and a great many representations and learning techniques have originated in this area. Consequently, while developments in the field have no *direct* application to Computer Audition, many of the general approaches (such as short-time spectral feature vectors, statistical classifiers, and time-series modeling techniques) are useful. Most relevant is work in noise robust speech recognition—recognition of speech in the presence of interfering noise [Moreno et al. 1996, Raj and Stern 2005, Wilson et al. 2008, Seltzer et al. 2004, Ellis et al. 2001]. Other related problems in speech recognition with strong analogues in Computer Audition occur in speaker diarization [Baokye et al. 2008a, Baokye et al. 2008b, Imseng and Friedland 2010]—the task of segmenting recorded conversations by speaker—and speaker recognition [García-Perera et al. 2012, García-Perera et al. 2013b, García-Perera et al. 2013a, Friedland and van Leeuwen 2010]—the task of identifying a speaker in a conversation. The Gaussian Mixture Model based techniques for speaker recognition have also been applied to the detection of acoustic events in consumer-produced videos [Mertens et al. 2011b] as part of TRECVID MED.

Source separation covers a range of approaches for dealing with the ubiquitous overlap of acoustic signals, an issue that is particularly prevalent in natural audio. Classic approaches include blind source separation (BSS), most popularly performed by independent component analysis [Bell and Sejnowski 1995, Smaragdis 1998]. Though elegant, this approach generally requires multiple sensor recordings (microphones) and assumes spatially compact sound sources, ruling out most real-world recordings. Another classic approach is computational auditory scene analysis (CASA) [Brown and Cooke 1994, Wang et al. 2006], which takes inspiration from the experimental psychology of listeners' perceptions of sound scenes, but still with a focus on reconstructing isolated target sounds, frequently from artificially controlled mixtures.

The most popular domain for CASA and similar systems is mixtures of two voices (including the Grid corpus [Cooke et al. 2006]), which can be separated, for instance, by exploiting the pitch tracks of both target and interference [Weintraub 1985, Brown and Cooke 1994, Hu and Wang 2003], or by using a speech recognizer to search for “legal” speech sound sequences consistent with the mixture [Hershey et al. 2010]. When the speech is from unknown sources, recorded in poor acoustic conditions, and/or mixed with non-stationary noise with unpredictable characteristics, all these approaches will encounter great difficulty. The “latent variable analysis” (LVA) approach [P. Smaragdis 2012, M. Shashanka 2008, Smaragdis and Raj 2011, Raj et al. 2011, Smaragdis and Raj 2010, Mysore et al. 2010, Raj et al. 2010a, Raj et al. 2010b, Wilson and Raj 2010, Singh et al. 2010a, Smaragdis et al. 2009b, Smaragdis et al. 2009a] learns the natural building blocks of individual sound sources by modeling them as sparse, convolutive compositions of over-complete bases. Sources are separated by algebraically identifying the building blocks from individual sound sources in composite sound mixtures. This approach is the basis of many state-of-the-art source separation algorithms today and will probably play an important role in the new field.

After speech, the most intensively studied class of audio is **music signals**. Following a series of early attempts at direct transcription of music audio [Moorer 1975, Mellinger 1991, Maher and Beauchamp 1994, Goto 2001, Klapuri 2003], the past decade has seen a rapid growth in systems for extracting information from music audio known as “music information retrieval,” and exemplified by the annual formal MIREX evaluations [Downie 2008]. The tasks addressed have broadened to include classification of genre [Tzanetakis and Cook 2002], mood, etc., and transcription of key, time signatures, and chords [Sheh and Ellis 2003]. Surprisingly, the audio features used in speech recognition—namely, Mel-Frequency Cepstral Coefficients (MFCCs) calculated over windows of a few tens of milliseconds—have also proven very successful for artist, genre, and instrumentation classification [Logan 2000]. MFCCs, however, deliberately attempt to be invariant to pitch (fundamental frequency) content, so they have frequently been augmented by so-called “chroma” features [Fujishima 1999] which typically fold tonal energy in the spectrum into a twelve-semitone vector where octave information (e.g., the distinction between C4 (262 Hz) and C5 (524 Hz)) has been removed. Chroma features have been particularly successful for chord recognition and matching of alternative musical performances (“cover songs”). More music-specific features have been suggested for some other specific tasks (e.g., the “beat histogram” for rhythmic classification [Tzanetakis and Cook 2002]). The multimedia field has become active in the development of this field, especially after the release of the 1 Million Song Corpus.

A series of publications on music classification by artist, genre, and tag [Berenzweig et al. 2004, Mandel and Ellis 2005, Mandel and Ellis 2008]; transcription of chords, notes, and rhythm [Poliner and Ellis 2007, Ellis 2007, Sheh and Ellis 2003]; and modeling music preference and similarity [Ellis and Poliner 2007, Jensen et al. 2009, Bertin-Mahieux and Ellis 2011, Müller et al. 2011] have been published in recent years.

The history of **environmental sound classification** originated with efforts to distinguish speech from nonspeech (typically music). This approach has since been extended to larger numbers of “generic” categories including things like environmental and man-made sounds (e.g., surveys by Chachada and Kuo [2014] and Duan et al. [2014] and the many papers cited therein), or mixtures of more than one class [Zhang and Kuo 2001], as well as a number of special cases such as finding pivotal events in sports videos [Xu et al. 2003, Xiong et al. 2003], or recognizing gun shots [Valenzise et al. 2007, Pikrakis et al. 2008] and bird sounds [Bardeli et al. 2010, Potamitis et al. 2014].

However, much of the recent effort in this area has focused on building classifiers for clearly specified lists of sound classes using carefully collected training data: the thrust being on improving the classifiers, and on collection of the data itself. Little attention has been paid to how the sound classes themselves may be defined. For instance, the CHIL Acoustic Event Detection campaign [Temko et al. 2006] compared several systems on a task of identifying individual “meeting room” events such as footsteps and cup clinks in both staged and real scenarios. More recently, the Detection and Classification of Acoustic Scenes and Events (D-CASE) challenge [Giannoulis et al. 2013], sponsored by the IEEE, included evaluation of a few “office” events (cough, phone, etc.) and also a broader task of classifying 30-second excerpts into 10 scene categories (street, supermarket, etc.). Participants reported an interesting range of results. However, the task implies small, closed sets of events and environments with no path to scale up to unconstrained, real-world audio.

Interest in the task of classifying relatively long snippets of audio into event categories has also been driven by the recently completed US-based IARPA Aladdin program. The task here is to recognize categories such as *woodworking*, *doing a skateboard trick*, *parkour*, etc., from the audio. However, the task itself is coarse-grained, only requiring categorization of entire recordings, rather than generating finer description.

In other work, researchers have developed approaches to recognizing environmental sounds based on automatically discovering the atomic units that compose the sounds, and the patterns of repetition that characterize different semantic enti-

ties in it [Chaudhuri and Raj 2011, 2012, Chaudhuri et al. 2011, 2012, Kumar et al. 2012]. Researchers have investigated how a library of sound events may be built up through automated and semi-supervised analysis of co-occurrences of these patterns in sound recordings, and how semantic assignments may be made by matching them to text and other descriptors. Other work has also developed theoretical machine learning frameworks to enable and analyze sparse estimation and structure discovery [Bahmani and Raj 2013, Bahmani et al. 2011, Bahmani et al. 2016].

A related environmental audio task is “geolocation”—recovering geographic location from ambient sound recordings. Friedland at ICSI pioneered this task, using Flickr videos from the MediaEval dataset [MediaEval 2010], in which audio was one of the modalities investigated. The ICSI location estimation system, for example, was able to locate 38.2% of the videos to within 10 km of the actual geo-location embedded in their metadata [Choi et al. 2011]. The Flickr videos, a precursor to the YFCC100M corpus (see Section 2.3), contain audio tracks that are “wild,” and contain large amounts of crowd noise, traffic noise, music, wind noise, and other environmental sounds. Some researchers also attempted audio-based location estimation at the city-scale. Obvious cues to location are scarce in such recordings, but the equal error rates (EERs) achieved by the system are better than 30% [Lei et al. 2012]. A system on user-verification is similar to its work on city-verification, where the objective is to determine whether two Flickr videos have the same user uploader, based on the audio tracks of the videos [Lei et al. 2011].

2.3 Data for Computer Audition

The suggested field will be enabled by the recent availability of large and diverse audio datasets based on consumer-produced videos, for example, the Multimedia Commons [ICSI et al. 2015]. The Multimedia Commons initiative is an effort to develop and share sets of computed features and ground-truth annotations for the Yahoo Flickr Creative Commons 100 Million dataset (YFCC100M) [Thomee et al. 2016], which contains around 99.2 million images and nearly 800,000 videos from Flickr, all shared under Creative Commons licenses.

In contrast to standard deep learning approaches, however, Computer Audition does not merely seek to engineer classification systems, and as such needs to evaluate systems with a broader set of queries. We envision something like IBM’s Watson system applied to audio recordings. It is important to emphasize that the development of rich evaluation methods is itself a research challenge, and will be a core component of the new field.

Therefore, in parallel with algorithmic innovation, the field needs to research the best ways of annotating and gathering training and test data with the aim of building a publicly available dataset of acoustic data that allows the benchmarking of our Computer Audition approaches. In general, a start could be to consider two major types of queries: those that concern the description of a particular observed sound, and those that seek sounds that satisfy a particular description. The queries will be driven by the models that people develop. In cases where a specific generative model is used, one needs to specify queries concerning subsets of the parameters of the model (e.g., “How much did the object weigh?”, or “Did the object break upon impact?”, or “Find other examples of sounds by a similar object”). In cases where it is possible to discover structure in the audio signal without supervision, researchers can use queries that relate to the learned structure (e.g., “Find examples that are similar to this target sound”). Computer Audition could also use queries about standardly labeled categories to show that the systems remain competitive at narrowly defined classification tasks.

2.4 The Nature of Audio Data

The analysis of sound presents difficulties not present in other domains such as vision or language. Most of the foundations of machine learning and analytics are built on assumptions that are problematic when processing audio data. These problems can for the most part be traced to one unique attribute of sounds: the fact that they superimpose. Consider as an example a simple classifier. Whereas it is a reasonable question in computer vision to ask whether an image region should be classified as “sky” or “skin,” an equivalent question on audio data is not as meaningful. Due to the fact that sounds superimpose, the correct answer could be that a given instant should be classified as simultaneously containing multiple classes. In fact, with real-world data the expectation is that a target sound will almost never be recorded in isolation, but rather will be part of a mixture. This issue renders the traditional classification question of “Which class are we observing?” inappropriate. At best we might instead ask “How much of each class do we observe?” This distinction is crucial in answering many key questions with sounds. For example, a mechanical engineer doesn’t want to detect if there *is* sound coming from ball bearings, but rather *how much* of it there is (which directly maps to their wear condition). Similarly, a doctor doesn’t want to know if there *is* a fetal heartbeat among all the body sounds of the mother, but rather *how strong* that heartbeat is. Standard classification and detection methods are not properly equipped to answer these questions, which is a primary reason why audio analytics hasn’t seen the same explosive growth as other modalities.

Furthermore, human descriptions of sound may be qualitative (e.g., a noisy recording or movie trailer). There may be detailed description in terms of known acoustic phenomena (e.g., there was a squealing sound, followed by a pop, a scream, and barking sounds); or they may assign higher semantics (e.g., tires squealed and a shot was heard; someone screamed and the dog barked). They may be more at an event level (e.g., a game was in progress, and people cheered). The description may be generic (e.g., “We heard music”), or specific (e.g., “Mozart’s 40th was playing”), or based on characteristics (e.g., “The music featured a four-four beat in C major). The level of description is not unique, and any and all of these may be invoked based on the desired analysis or outcome. Also, descriptions may not always be at a semantic level, such as in the above examples. They will often be in terms of similarity (e.g., “I’ve heard this sound before”), or context (e.g., “This sound generally follows hammering in these recordings”), or even abstract (e.g., “Something about this sounds familiar”). An ideal automated system will be able to perform or support all of the above modes. But beyond simple detection or identification of known concepts is the inference of what was previously unknown—e.g., “Oh! So this is what a zebra sounds like—from other information, such as that this recording contains a zebra. While as listeners we are largely unaware of the sophistication of the analysis and inference we perform on the sounds we hear, the analysis and inference present a very challenging set of tasks to be addressed by machines.

We need to differentiate between acoustic backgrounds and acoustic events on the basis of their temporal extent. Acoustic backgrounds such as speech, music, rain, wind, crowd noise and traffic noise, consist of sounds that generally last for longer durations in the video; they may not be background in the sense of being of secondary interest, but they will typically constitute a background context for shorter-duration acoustic events that may overlap with them. Acoustic events are associated with isolated noises of short duration, such as door slams, car horns, thunder, jet engine noise, bird noise, and footsteps. Listening to the soundtracks suggests that much of natural audio revolves around these fundamental acoustic backgrounds and events, and being able to automatically discern them via automatic means is the goal of this field.

2.5 Dealing with the Peculiarities of Sound

The focus of the field will be on combining methods from the audio communities involved in speech recognition, music analysis, cognition, machine learning, and multimedia retrieval and then adopting them for environmental audio. We view the purpose of audition as that of recovering the latent structure in audio to facilitate

behavior and cognition. The approaches we will take to developing Computer Audition thus in various ways all involve inferring or discovering latent structure from sound. In some cases this structure consists of the distinct sources that concurrently generate an audio signal. In others it consists of physical properties that together determine what we hear. And in others it consists of discrete events embedded in a continuous audio signal. The experimental systems described below each address a different type of latent audio structure. We view these as the first baby steps researchers could take toward initiating the new field.

2.5.1 Representation and Parsing of Mixtures

Noise removal and source separation have been studied for decades. The goal of such methods is to suppress unwanted sounds in order to simplify further processing. Although such methods have found moderate success in some areas (e.g, ambient noise reduction for voice communication), they are not as helpful when performing audio analytics because they rely on a target vs. noise model of processing and are poorly suited to explaining ambient sound scenes or complex mixtures with no clear foreground and background.

An alternative to suppressing unwanted elements is to use representations that facilitate inference in the presence of mixtures. A popular approach that makes use of this philosophy is that of *compositional models* of sound [Virtanen et al. 2015]. Such models represent audio recordings as additive compositions of dictionary elements and do not make use of cross-cancellation. As a result, they describe sounds in a manner very similar to how we think of them: as a sum of parts. Such models have proven successful at describing complex sound mixtures in a way that allows computational inference on each of the constituent sources separately (e.g., simultaneous pitch tracking of multiple instruments in a piece of polyphonic music).

Current approaches using compositional models are dependent on multiple decisions by the user, such as the domain where the dictionary elements are defined, the shape of the dictionary, the invariances of the dictionary elements, priors on the usage of these elements, etc. As computational inference has entered a new era, in the last few years we have found that one can leverage large amounts of data to bypass the making of such design considerations and instead rely on data-driven decisions. Taking advantage of our audio data collection effort, our intent is to explore fully data-driven compositional model design. The advantage relative to previous methods will be to produce compositional models that are arbitrarily customizable for different domains. Consider, for example, two researchers, one working on musical signals, the other working on insect sounds. Current audio

tools are very well equipped to work with music because it is a domain that has been extensively investigated and one in which we know how to pick processing parameters that work best. Knowing that pitch is important in music, we would pick suitable large processing windows; we would know that a constant-Q front-end transform would represent musical structure well, and that our dictionary sizes should be roughly in the order of music notes that we expect to encounter. However, these design decisions would be catastrophic in the case of insect sounds, which are not as pitched, exhibit very fine temporal resolution, and can't be easily described in terms of known sound units (like notes, or phonemes). In the latter case we need systems that learn the optimal representation, as opposed to looking for guidance from a user.

Previous work on fully data-driven compositional systems is pointing in this direction. Using a very flexible and abstract formulation of the compositional pipeline in the form of a deep multilayer network, we can construct sound models that are completely tuned to the sound classes they are designed to work with, promising optimal performance with no parameter guessing by the user. This approach will not only allow us to learn the most general models based on the annotated data above, but it will also allow domains specialist to take such tools and customize them for the specific sound domains they are working on.

Traditionally, compositional models require various user design choices (e.g., type of front-end, size of dictionary, representation of signal, etc.). This often results in systems that can work well in one audio domain (e.g., speech), but fail spectacularly in others. The success of such models often hinges on the user's intuition in picking the right structure.

2.5.2 Grouping Audio Segments

Another major challenge with audio is finding the boundaries of when one sound begins and ends. We believe one of the first steps in developing Computer Audition is formulating and answering questions like: How many sounds are in the audio piece? Are there overlapping events? When does a particular audio event end and when does it start? In other words, what's the smallest chunk of sound in an audio file that users would perceive as "belonging" to the same sensation of sound? For example, humans readily recognize both a "gunshot" or a "siren," even though these two sounds typically have completely different durations.

In previous work based on speaker diarization [Mertens et al. 2011a], researchers have shown that grouping similar audio chunks (which we call "percepts") can help with video retrieval [Elizalde et al. 2012]. However, these experiments didn't take acoustic or perceptual models into account; they were pure

applications of machine learning. Having said that, the answer to these questions is not always clear-cut, as sound events can overlap or be intentionally mixed. As a first step toward using models to segment audio, the field should investigate the application of perceptual similarity metrics, i.e., sounds that have different causes can be grouped together if they match perceptually. In further stages, researchers might also incorporate semantic similarity via multimodal context (e.g., visual input or metadata). But even perceptual matching alone is potentially important, as it allows for the fundamental operation of comparison, which in turn enables searching and, to a limited extent, sorting. Note that text-based search and string operations also mostly occur without the notion of semantics.

2.5.3 Learning and Exploiting the Generative Structure of Audio

Although the explosion of deep learning has led to great success in classification tasks in many different domains, the resulting systems are by nature classifiers. Current automated descriptions of audio are generally in terms of identifiable “familiar” events, and are thus notably limited by the particular sets of labels they are trained on. If a system is trained to recognize the sound of a car, with enough training it will produce that label for signals that resemble those labeled as cars in the training set. However, if a user is interested in knowing whether the car was driving fast or slow, there is no way for the system to provide an answer without retraining it on a new dataset. Interpreting and making inferences from sounds requires the ability to describe their structure in a manner that permits additional reasoning and inference.

One shortcoming of conventional supervised learning approaches is that they do not leverage or learn the processes that generate signals. Human listeners, in contrast, are frequently able to infer what happened to cause a sound, enabling rich and flexible inferences. We realize not only that someone is walking outside our office, but that they are approaching, and then pause, and walk away. We learn that machines produce noise, but also that the spectral properties of the noise vary with the speed of the engine. We can recognize that it is raining, but also whether it is raining hard or drizzling, whether the rain is falling on a lake or a tin roof, and whether the wind is gusting at the same time. All of these judgments plausibly depend on generative knowledge of the processes that produce sound and the parameters by which they vary. We suggest instantiating machine systems that can similarly learn generative structure and use it to infer the causes of sound in the world. The advantage of this approach is that a system can then be queried about any parameter or combination of parameters in the generative model, and

does not have to be retrained any time a user seeks to ask a different question about what happened to cause the sound.

2.5.4 Leveraging Generative Models of Sound

Impact sounds are one domain in which the physical processes that produce sound can be specified in detail, and constitute a useful test case for generative models in hearing. State-of-the-art sound synthesis can render audio from parameters of objects and the surfaces that they impact (shape, mass, material) along with parameters of the object motion (speed and trajectory). Depending on the material, shape, and motion, objects can bounce, roll, or shatter. The trajectory, and the sound it produces, depends on the geometry of the surface(s) they impact. Humans are adept at inferring the latent physical variables from listening to the sound that is produced, and we suggest producing machine systems with similar capabilities using physical generative models.

Vocal sounds are another domain in which inference in generative models holds promise. Recognition via production models is an old idea in speech, but has never taken hold in machine speech recognition, in part because of a lack of training data, and in part because the inference of articulatory parameters from audio is ill-posed and non-convex. We do not advocate using articulatory models for speech recognition, as it is currently being handled effectively by data-driven deep learning approaches, but nonetheless think it has promise for understanding vocal sounds more generally. Humans produce many vocal sounds that are communicatively important despite not conforming to the typical definition of speech, as when we sigh, moan, groan, giggle, or cough. Each of these types of sound varies depending on the speaker's mood or intention. A generative approach could recover the vocal parameters needed to synthesize an observed sound, and would likely be useful in estimating communicative intent.

The advantage of inference in a generative model is that the system can be queried about whatever component of the generative process is task-relevant. In some situations it may be important to know when an object that was dropped made its first impact. In others it may be important to know what the object was, or what material it was made from. Rich generative models have the ability to be queried on any of the possible parameters that determine the sound they produce. They thus enable the flexibility that is an essential aspect of perceptual intelligence. Other domains in which similar approaches will be explored include vocal sounds (using articulatory models) and human action sounds (using models of rhythmic motor movements).

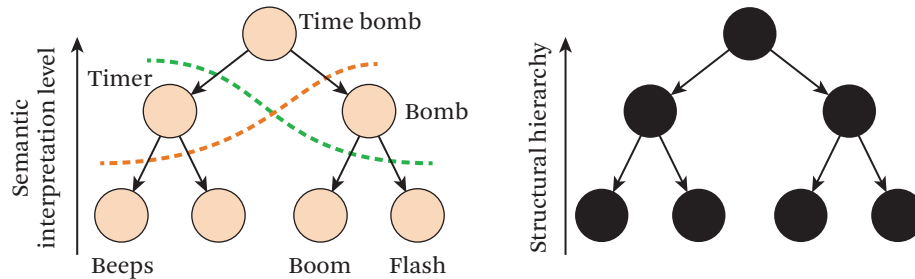


Figure 2.3 Left: Semantic hierarchy for a “time-bomb.” Dotted lines represent two different descriptions that a person might provide. Each of them “cuts” the tree differently. Right: An attempt to recover the entire structure without supervision.

2.5.5 Structure Discovery in Audio

Continuing the above theme, most current approaches for the analysis of audio are *shallow*—they either learn the overall distribution of the data [Pancoast and Akbacak 2011, Zhuang et al. 2011], or perform spot-identification of the events contained, without inferring any higher-level structure. The problem is illustrated by Figure 2.3. Two subjects describing the same event—the explosion of a time bomb—might describe it as “there was a series of beeps, and a bomb went off,” or “a timer ticked down, and there was a boom.” Although contrived, the example illustrates a problem: The semantic structure of the audio is hierarchical, and different descriptions of the same episode may “cut” the tree at different levels, giving very different results. Recognizing and reconciling the different descriptions of the same episode can be challenging.

On the other hand, if the entire tree is specified instead of just a cut, descriptions are unambiguous. It also becomes possible to reconcile *different* recordings of the same episode, since they may be expected to have identical trees. Recordings that represent *similar* episodes can be identified based on the similarity of subtrees, etc. Unfortunately, a complete description of the entire tree would require semantic labeling of every level of the hierarchy: infeasible given the unlimited number of labels possible.

In a series of articles, Chaudhuri and colleagues suggest that a reasonable alternative is to extract the entire hierarchy in an *unsupervised* or *semi-supervised* manner [Chaudhuri and Raj 2012, Chaudhuri et al. 2012, Chaudhuri 2013]. The suggested model assumes a simple structure: the lowest layer of the tree comprises low-level sounds (e.g., beeps and clicks), patterns over these low-level sounds rep-

resent higher-level concepts (e.g., a timer), patterns of higher-level concepts form still-higher-level “episodic” concepts, and so on. We show how all of these levels may be derived in an unsupervised manner by assuming a generative model with a language-like structure [Chaudhuri 2013, Walter et al. 2013] characterized by a power-law distribution and low perplexity. The recovered structures provide a viable handle for cataloguing, comparing, and retrieving recordings, enable inferences that can deal with ambiguous definitions, and even permit extractive summarization of the audio by retaining only the most relevant or salient segments.

Despite its success, the model suggested by Chaudhuri and colleagues is still rudimentary [Chaudhuri and Raj 2012, Chaudhuri et al. 2012, Chaudhuri 2013]. It works entirely from *surface* information, namely the audio, with minimal reference to external information about the structure of the data, and does not recover generative models that capture the structure of the actual underlying physical generative processes. As such, it can only be considered a proof of concept. To facilitate more comprehensive analysis and inference, we suggest scaling up the model to deal with large volumes of audio and with continuous learning. The models will then incorporate weak external supervision at various levels to constrain the learned tree (e.g., knowledge about the occurrence of individual known events or the category of the recording, information that different recordings represent the same episode, information derived from external labels or metadata, etc.). Those in this field should also investigate other statistical models, and draw upon techniques developed for computational modeling of language and computer vision, both of which provide computational analogies to the approach discussed here [Niebles et al. 2008, Tang et al. 2009, Sankaran 2010].

2.5.6 Learning to Describe Sound: NELS

Higher-level inference regarding sounds, even after recovery of acoustic structure, requires describing sounds in a form that permits inference. A fundamental characteristic of sound is that it is the result of *actions* or *interactions* of objects. This results in a virtually unlimited number of ways of describing sounds. They may be described through simple direct descriptors, for example, by words such as “bang” or “miaow”; through descriptive phrases such as “cat sound”; or, more abstractly, through action phrases such as “car skidding into a wall” where none of the words individually signifies a sound, but collectively they form a phrase that could. Thus, simply knowing when a description alludes to a sound, or, conversely, determining when a recorded sound is distinctive enough to merit assignment of a name or description, is a challenge in itself. In prior work, Kumar et al. [2014] have attempted

to obtain a phenomenological definition of name-worthy sounds. More recently, researchers have explored the automatic identification of sounds that derive from logical composition of actions and objects [Saeger et al. 2016, under review].

A more complete proposal, which encompasses these and much more, is a system we refer to as the “Never-Ending Sound Learner” or NELS [Elizalde et al. 2016]. Inspired by Carnegie Mellon’s “Never Ending Language Learner” (NELL) system, NELS is a semi-supervised system that learns to identify and group valid descriptions of sound and, where possible, associate them with examples by analyzing vast quantities of audio and multimedia data and their associated metadata and surrounding text on the web and other repositories. The system builds on information derived from existing ontologies or knowledge bases such as those derived by the CMU NELL system. The key to the functioning of the system is that, given the sheer volume of data, it can operate in a high-precision low-recall regime: although the system may miss the vast majority of instances of any sound, we can be certain about the sounds and sound-related facts it does discover. NELS will identify known sounds in recordings; detect the occurrence of new, previously unknown sounds; then learn to recognize them and associate names with them to increase its vocabulary. It will also learn physical and common-sense structural and temporal relationships between sounds, plus audio-visual associations and associations between sounds, their meaning and semantics and categories. The system is intended to run continuously, seeking out new information and data, expanding its own knowledge base, and refining existing knowledge. In time, it is intended that the system will provide the largest and most up-to-date available repository of sounds, and a complete ontology of sounds, all automatically derived.

2.5.7 Learning from Weak and Opportunistic Supervision

Eventually, learning to identify or recognize the various component sounds that occur in an audio segment, either by name or by other characteristics that can be described, requires labeled examples. Labeling is an expensive, tedious, and time-consuming process. On the other hand, *weak* labels, which only provide approximate information about the composition of sound recordings, are much easier to obtain. For instance, it is much easier to label the presence or absence of a specific sound in continuous recordings than it is to mark the actual locations where it happens. In prior work [Dietterich et al. 1998, Maron and Lozano-Perez 1998], we have shown how such weak labels can be used to train reliable sound classifiers/detectors through techniques based on multiple-instance learning. In pilot experiments, researchers were not only able to build classifiers in this manner,

but also automatically derive additional label information about the number of instances of the sound and their timing.

More generally, weak labels can be gathered in large quantities from web sources such as media sharing sites like Flickr and YouTube. These sites host audio and multimedia recordings, many of which have loose annotations in the form of metadata. Others are accompanied by user comments, which relate to the content of the audio at least in some cases. All of these represent “casual” annotation, which may (or may not) indicate the presence of sounds, and other information such as the number of occurrences, similarities, repetitions, or other abstract indicators.

2.6 Potential Applications

Effective computer listening systems will have widespread applicability. Relevant domains include ecological monitoring (e.g., wildlife population estimates through sound), medical diagnoses (e.g., automated auscultation), mechanical monitoring (e.g., prediction of malfunction in heavy machinery via sound), surveillance applications (e.g., detection of automotive accidents via microphones), and Internet retrieval (see Figure 2.4). Unlike other data-rich fields, such as vision and language, we currently lack tools to allow a non-expert to mine large sound databases. Specialized tools often focus on narrow domains, such as speech or music, but they do not generalize to other types of sounds and are typically not useful outside their narrow scope. The goal of Computer Audition is to provide the groundwork for general audio analytics systems and to make these tools available to a wide range of relevant fields.

2.7 Conclusion

This chapter has presented a proposal for a new field of scientific endeavor: Computer Audition. We argue that because of past work and the enabling datasets, this field fits well within the multimedia community and therefore presents one of its frontiers. We have outlined initial steps and ideas to solve some of the challenges that will be met immediately when trying to achieve progress. Moreover, this chapter has only described Computer Audition as a singular field. It is well known, however, that the human brain integrates visual information with context and memory as well as other sensors when performing the task that is widely known as listening. Multimodal integration would therefore be a very important focus of this endeavor. Ultimately, this chapter presents but a small initial view of what the real work will look like once brave researchers start to be pioneers on this frontier.

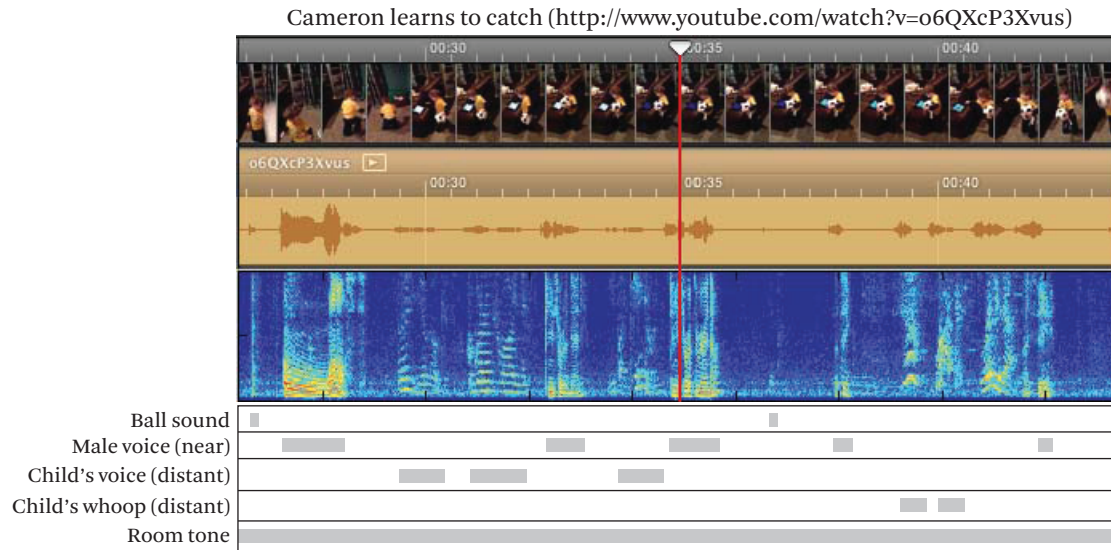



Figure 2.4 One of the many immediate applications of Computer Audition: understanding and modeling soundtracks of consumer-produced videos, facilitating retrieval, automatic editing, and transcription. These functions could be transformative: Improved retrieval, for instance, would enable field studies of never-before-seen scale, e.g., on how babies learn to catch a ball.

Nevertheless, we hope that this chapter can serve as a constructive inspiration and guide for the initial baby steps toward making computers listen to the world.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1629990 to Gerald Friedland, NSF award #1453104 to Paris Smaragdis, and a McDonnell Scholar Award to Josh McDermott. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or other sponsors. We thank Daniel P.W. Ellis for the many discussions on these ideas, his input on past work, and Figure 2.4.



Multimodal Analysis of Free-standing Conversational Groups

Xavier Alameda-Pineda (Inria),
Elisa Ricci (University of Perugia),
Nicu Sebe (University of Trento)

“Free-standing conversational groups” are what we call the elementary building blocks of social interactions formed in settings when people are standing and congregate in groups. The automatic detection, analysis, and tracking of such structural conversational units captured on camera poses many interesting challenges for the research community. First, although delineating these formations is strongly linked to other behavioral cues such as head and body poses, finding methods that successfully describe and exploit these links is not obvious. Second, the use of visual data is crucial, but when analyzing crowded scenes, one must account for occlusions and low-resolution images. In this regard, the use of other sensing technologies such as wearable devices can facilitate the analysis of social interactions by complementing the visual information. Yet the exploitation of multiple modalities poses other challenges in terms of data synchronization, calibration, and fusion. In this chapter, we discuss recent advances in multimodal social scene analysis, in particular for the detection of conversational groups or F-formations [Kendon 1990]. More precisely, a multimodal joint head and body pose estimator is described and compared to other recent approaches for head and body pose estimation and F-formation detection. Experimental results on the recently published SALSA dataset are reported, they evidence the long road toward a fully automated high-precision social scene analysis framework.

3.1 Introduction

Several wearable sensing devices became available for the general public over the past few years. Often, such consumer platforms include several sensors (e.g., microphone, accelerometer, or Bluetooth). Investigating methods to process the data gathered with wearable devices is worthwhile for many reasons. First of all, considering that modern consumer platforms, such as smartphones or smartwatches, are within reach of the general public, many are already using them for data acquisition. Second, these platforms are inherently multimodal, thus they provide a rich description of the environment. Third, because these devices are easy to wear, they can be used in casual, real-life situations, far from laboratory-controlled conditions: this makes the continuous flow of information emanating from them a precious resource for studying these settings.

Importantly, wearable technologies are complementary to distributed sensor networks. For example, when analyzing social interactions, wearable sensing devices can be exploited *inter alia* to localize people and to roughly estimate their activities. However, a fine-grained analysis of the social scene requires additional information gathered with alternative distributed sensing networks, e.g., visual data [Pantic et al. 2005]. Developing methods that robustly fuse data from wearable devices and distributed networks remains largely unexplored, and many challenges arise in this context. As with any consumer device, data gathered with wearable technology can be corrupted by severe noise, and often some extra processing is required to remove it [Yatani and Truong 2012]. Therefore, unimodal approaches fail to provide a robust and accurate representation of the environment and smart multimodal fusion strategies need to be developed [Lingenfelter et al. 2014]. Moreover, these strategies should also account for the specificities of the combination of a distributed sensor network and the wearable technology.

The focus of this chapter is the analysis of spontaneous social interactions in natural indoor environments (see Figure 3.1). In such social events, human beings tend to organize themselves in free-standing conversational groups (FCGs). Contrary to the staticity proper to round-table meetings, FCGs are dynamic by nature (they increase and decrease in size, move and split) and therefore their analysis is inherently difficult [Cristani et al. 2011, Setti et al. 2013]. Indeed, such scenarios are particularly difficult because people are moving around, engaging in and quitting conversations, etc. In this context, robust human pose estimates can ease the further mining of higher-level descriptions, such as F-formations or attentional patterns. Therefore we focus on the estimation of the human pose of several persons in a crowded regular indoor environment. Clearly, this is inter-related with other tasks such as multi-person tracking [Ba et al. 2016], activity



Figure 3.1 People interacting, naturally organized into free-standing conversational groups, during a poster session. The social interplay is captured by a distributed camera network and by sociometric badges worn by each participant (small white boxes).

recognition [Chéron et al. 2015], sound source separation [Kounades-Bastian et al. 2016], and diarization [Kounades-Bastian et al. 2017] (who speaks when). However, all these problems and their dependencies are out of the scope of the chapter.

Providing a complete description of the human pose (positions of the limbs) is chimeric in video surveillance settings, i.e., when people are monitored with distant and large field-of-view cameras. Indeed, the low-resolution images and the numerous occlusions have a relentless negative effect on current approaches. Fortunately, the head and body pose can be used as a surrogate for the full pose when analyzing free-standing conversational groups. Indeed, *all* features delineating group social interplays are often extracted from head and body pose estimates (HBPE). More precisely, accurate and robust HBPE are the prelude to detect face-to-face interactions, identify the potential addressee and addresser, describe the geometric structure of the group (or F-formation), and recognize personality traits. Many research studies focused on HBPE from visual data in the recent past, and demonstrated the synergistic effect of joint head and body pose estimation, when compared to independent estimation [Chen and Odobez 2012]. In addition, we are expressly interested in analyzing FCGs using head and body pose estimates obtained through the processing of the multimodal flow of data emanating from a distributed camera network and sociometric badges [Choudhury and Pentland 2003] worn by

each participant. Even if the community invested lots of effort in the estimation of head and body poses from visual data, very few researchers focus on how to jointly estimate both of them from data gathered with multiple different (distributed and wearable) sensors.

In this chapter we propose to integrate multimodal signals acquired in regular indoor environments to robustly extract HBPE of several people naturally arranged in FCGs. More precisely, we use auditory and proximity (infrared) information to automatically label visual features of head and body, respectively. All this information, together with the visual features extracted from the surveillance camera network, are pulled together into a linear classification problem. Seeking for the non-available labels, i.e., estimating the head and body pose of all people, is cast into a matrix completion problem. Matrix completion (MC) is a methodological choice motivated by mainly three factors. First, even if the original matrix completion problem is NP-hard, relaxations exist so that the optimization is cast into an alternation of convex problems, leading to computationally efficient algorithms. Second, MC lies within the class of transductive methods, meaning that all the information (including the features of the test set) is used to train the classifier, thus further regularizing the training process. Finally, matrix completion is, by nature, able to deal with missing labels/features, which makes an optimal candidate for analyzing crowded scenes recorded with wearable sensors. Indeed, crowded scenes imply visual occlusions and wearable sensors are often intermittent, leading to much data missing in the incoming flow of observations. Furthermore, within this framework, the joint head and body pose estimation adds an additional coupling term to the formulation. Similarly, the temporal structure inherent to the problem is induced by means of a Laplacian matrix that regularizes the matrix completion task. The resulting HBPE are then used to infer the F-formations.

With this chapter we aim to: (i) open up the doors of multimodal information fusion to a problem that has kept the computer vision and pattern recognition communities busy (and still does), (ii) evaluate under which circumstances the use of multimodal data emanating from wearable sensors can enhance the estimation, and (iii) give general guidelines for the design of new methodologies able to process these hybrid distributed-wearable data streams.

The rest of the chapter is structured as follows. Related approaches for head and body pose estimation are discussed in the next section. The proposed matrix completion model, accounting for noise, temporal regularization, and joint head and body pose estimation, is presented in Section 3.4, together with the associated optimization algorithm. Extensive evaluations of the proposed approach, and com-

parisons with the state of the art, are reported in Section 3.5, before concluding and delineating future research directions in Section 3.6.

3.2 Related Work

The applicative context of the present study is the multimodal analysis of social interactions in informal gatherings. In this context we focus on the estimation of the head and body pose, facilitating the automatic study of free-standing conversational groups. Since the problem is cast into a matrix completion framework, we also give an overview of that topic.

3.2.1 Multimodal Analysis of Social Interactions

Combining data from heterogeneous modalities is of utmost importance for understanding human behaviors [Song et al. 2012]. Approaches relying on audio-visual cues are probably the most popular and successful examples [Alameda-Pineda et al. 2013, Gatica-Perez 2009, Petridis et al. 2013, Vinciarelli et al. 2009, Alameda-Pineda and Horaud 2015, Ba et al. 2016, Gebru et al. 2016, Ricci et al. 2013]. In Gatica-Perez [2009], the role of non-verbal cues for the automatic analysis of face-to-face social interactions is investigated. Social interactions in political debates are studied in Vinciarelli et al. [2009]. A multimodal approach for detecting laughter episodes from audio-visual data and a novel dataset for studying this problem are introduced in Petridis et al. [2013]. In the last few years, mobile and wearable devices have opened novel opportunities for multimodal analysis of social interactions [Campbell et al. 2006, Eagle and Pentland 2006, Do and Gatica-Perez 2013, Matic et al. 2012, Alameda-Pineda et al. 2015]. In Do and Gatica-Perez [2013], a probabilistic approach is proposed to automatically discover interaction types from large-scale dyadic data (e.g., proximity Bluetooth data, phone call network, or email network). In Matic et al. [2012], social interactions on a small spatio-temporal scale combining proximity data and accelerometers are analyzed.

On the one hand, traditional distributed camera and microphone networks are able to capture subtle features of human behavior, but their performance drops significantly when dealing with a crowded scene. On the other hand, wearable devices permit the ubiquitous localization of people for a long time span, as they typically embed proximity sensors, but they fail to provide accurate and detailed descriptions of the ongoing interplay. Table 3.1 provides a short description of the advantages and disadvantages of using each of the wearable sensors. Therefore, it is obvious that the nuances and the complexity of social interactions require

Table 3.1 Advantages and disadvantages of different wearable sensors

Sensor	Advantages	Disadvantages
Proximity (infra-red)	Robust and reliable. They can be used for synchronizing the badges to the camera network as in Alameda-Pineda et al. [2016b] .	Very limited amount of information. Not too accurate.
Audio	Rich amount of information with high descriptive potential.	All information is mixed in a highly non-stationary signal. Blind processing is extremely challenging.
Accelerometer	Provides information about the activity and dynamics of the person to some extent.	Very noisy signal; difficult to exploit as standalone.

leveraging information from both wearable and traditional sensors. This chapter develops from this intuition and, to our knowledge, it is the first work jointly employing sociometric badges (see [Figure 3.2](#)) and external cameras for head and body pose estimation.

3.2.2 Head and Body Pose Estimation

As already outlined, the analysis of human behavior will definitely benefit from automatic human pose estimation. Multimodal approaches based on RGB-D data [[Shotton et al. 2013](#), [Yan et al. 2014](#)], eventually combined with audio [[Escalera](#)



Figure 3.2 The sociometric badge used in our experiments. (From [Alameda-Pineda et al. \[2015\]](#))

et al. 2014], and with visual-inertial sensors [De la Torre et al. 2009], have recently proved very successful in tracking human limbs. However, the former methods are appropriate when only one or few people move in proximity to the camera, while the latter approaches are often not practical solutions, as they require the person’s body to be equipped with multiple sensors. In addition, the robust estimation of the full human pose is chimeric in crowded scenes, which is typically the case in informal social gatherings. Consequently, when considering social interactions among several people, the head and body pose can be used as a surrogate for the full human pose.

Previous research has demonstrated that head and body orientations can be successfully detected from visual data [Benfold and Reid 2011, Chen and Odobez 2012, Yan et al. 2013, Rajagopal et al. 2014, Rajagopal et al. 2012] and used as primary cues to infer high-level information, such as visual focus of attention [Voit and Stiefelhagen 2010] and conversational groups [Cristani et al. 2011, Ricci et al. 2015]. In order to aid the human annotation effort, previous research has focused on using related cues to estimate head and body pose. For example, Benfold and Reid [2011] consider head pose labels generated using walking direction. Similarly, coupling of head and body pose due to anatomical constraints is reinforced by Chen and Odobez [2012]. Body orientation is considered as a link between walking direction and head pose in Krahnstoeber et al. [2011] and Robertson and Reid [2006]. Rajagopal et al. [2014] propose a transfer learning framework for head pose estimation under target motion. In Yan et al. [2013], a multi-task learning approach is introduced for accurate estimates of the head pose from large field-of-view surveillance cameras. More recently, approaches to specifically cope with label noise [Geng and Xia 2014] and integrate temporal consistency [Demirkus et al. 2014] have been introduced. In their 2008 paper, Canton-Ferrer et al. likewise show their efforts toward audio-visual head pose estimation. However, to the authors’ knowledge, there are no research studies combining visual features from a distributed camera network with multimodal cues extracted from wearable sensors to jointly estimate head and body poses.

3.2.3 Matrix Completion

We propose to cast multimodal head and body pose estimation into a matrix completion problem, which has been shown to be equivalent to learning a classifier in a transductive setting [Goldberg et al. 2010]. This formulation is particularly advantageous when data and labels are noisy or in the case of missing data. Data can be noisy, for instance, when the scene is crowded. Indeed, occlusions will occur, and feature-extracting algorithms may then provide features that do not correspond

to the right person. Similarly, wearable sensors can be inaccurate or provide mixed information for two or more people at the same time. In the computer vision and multimedia communities, this fact has been exploited in several applications, such as multi-label image classification [Cabral et al. 2014, Alameda-Pineda et al. 2016a], image retrieval [Wu et al. 2013], and facial analysis [Wu et al. 2015a, Tulyakov et al. 2016]. A recent study [Kalofolias et al. 2014] extends the matrix completion problem to take into account an underlying graph structure inducing a weighted relationship between the columns and between the rows of the matrix. We also utilize this structure to model the temporal smoothness of the head and body pose estimates. However, from the technical point of view our approach is novel because: (i) we address two matrix completion problems (for the head and body poses) that are coupled and cannot be reduced to a joint matrix completion problem, and (ii) the proposed framework is able to deal with data coming from multiple modalities, handling signals generated from different types of sensing devices (distributed and wearable).

3.3 The SALSA Dataset

The technical challenge of the present study is the robust estimation of the body and head pose in crowded scenarios for further analysis of FCGs. In particular, we use the SALSA (Synergistic social Scene Analysis) dataset, which records a two-part social event involving a fixed number ($K = 18$) of participants. While the first half consists of a poster presentation session (see Figure 3.1), the second half consists of a cocktail party with food and beverages. The scene of SALSA is captured by a distributed camera network and by sociometric badges worn by the participants. Specifically, visual data are recorded by four synchronized static RGB cameras operating at 15 fps. The sociometric badges are equipped with a microphone and an infrared (IR) beam and detector. Importantly, these wearable devices are battery-powered and store the data on a USB card without the need for any wired connection, thus guaranteeing a natural social interplay. SALSA also comprises manual annotations every 3 seconds of position, head and body orientation of each target, and F-formations.

Our aim is to estimate the head and body pose of person k at time t , denoted by θ_{kt}^h and θ_{kt}^b , respectively, for all k and t . Different possible representations exist for the body and head orientations; inspired by the literature [Chen and Odobez 2012], we sectorize the orientation space in the plane into $C = 8$ classes. In other words, we consider each of the poses as a unit vector in the plane belonging to 8 possible classes $\{[0^\circ, 45^\circ), \dots, [315^\circ, 360^\circ)\}$. Therefore, the body and head orientations are C -dimensional vectors. $\left(\theta_{kt}^b\right)_c$ should be understood as the probability that the

associated body visual feature lies in the c^{th} sector (analogously for the head). For instance, $\theta_{kt}^b = [1, 0, \dots, 0]$ means that θ_{kt}^b belongs to the first sector. At training time, these will be either manually annotated, automatically extracted from the badge's information, or unknown. More precisely, the infrared and auditory information from the sociometric badges are processed in order to automatically extract body and head pose labels, respectively. Indeed, the different nature of the infrared and auditory signals carries a rich representation of the badge's owner with respect to the other participants. On one side, if the badge of person k detects the infrared beam of person l , most likely k 's body orientation points to l , since the badge is worn on its owner's chest. On the other side, if the auditory signal of k 's badge is highly correlated with the one of l , it is likely that l is speaking toward k , and that therefore the head of l points toward k . This hypothesis could be violated when two people are discussing something close to each other, since both speech signals would be recorded by both microphones and their heads would not necessarily be pointing toward each other. It is worth noticing that, while visual features are extracted in a continuous manner, infrared and auditory labels are sparse.

In order to estimate head and body orientations, we need to extract visual features from the videos recorded by the camera network. First of all, the participants are tracked on the visual stream with multi-target tracking algorithms with state-of-the-art occlusion handling such as that in Lanz [2006]. The estimates of the ground positions are used together with the camera calibration to retrieve the bounding box of the head and body of all participants for each camera view. We use those frames in which all participants are seen from all four cameras. We then choose to describe each bounding box with histogram of oriented gradient (HOG) descriptors [Dalal and Triggs 2005]. More precisely, the head and body images are first normalized to 20×20 and 80×60 pixel images, respectively, from which we extract HOG in non-overlapping cells of 4×4 pixels. Finally, the descriptors for all four views are concatenated, and the dimensionality of the head and body descriptors is reduced using principal component analysis, keeping 90% of the variance. In all, the head and body visual features extracted from the four cameras for each participant ($\mathbf{v}_{tk}^b \in \mathbb{R}^{d_b}$ and $\mathbf{v}_{tk}^h \in \mathbb{R}^{d_h}$) are roughly 100-dimensional.

3.4 Matrix Completion for Multimodal Pose Estimation

We assume the existence of a labeled training set \mathcal{L} containing the raw features and noisy labels of the head and body pose of all K persons involved in the interaction for $T_0 < T$ frames, where T is the total number of frames. That is: $\mathcal{L} = \left\{ \mathbf{v}_{kt}^b, \theta_{kt}^b, \mathbf{v}_{kt}^h, \theta_{kt}^h \right\}_{k=1, t=1}^{K, T_0}$. Complementary to this, the set of unlabeled features is

defined as $\mathcal{U} = \left\{ \mathbf{v}_{kt}^b, \mathbf{v}_{kt}^h \right\}_{k=1, t=T_0+1}^{K, T}$. It is important to remark that part of the training set is manually annotated and the rest is automatically labeled. More precisely, we use the infrared detections as a proxy for the body pose and auditory correlations to automatically label the head pose. Indeed, it is reasonable to assume that the audio signals have a strong influence on the gaze direction of the targets, while the infrared detections naturally correlate with the body orientation. Finally, it is worth noticing that while visual data is continuously available, audio and infrared signals are sparse observations. In the following we describe the proposed approach for fusing these multiple modalities, so as to robustly estimate the head and body poses of all participants.

3.4.1 The Model

Recent theoretical results and practical advances in matrix completion theory [Cabral et al. 2014, Goldberg et al. 2010] motivated us to consider the head and body problem from this perspective. We thoroughly cast the estimation into a matrix completion problem, thus introducing matrix completion for head and body pose estimation (MC-HBPE), a graphical illustration of which is shown in Figure 3.3. In the following, in order to facilitate the exposition, all the claims, definitions, and properties are stated only for the body pose, but they remain true for the head pose, unless it is explicitly written otherwise. Before describing the model, we set a few useful notations: the matrices of labeled and unlabeled body features of person k , $\mathbf{V}_{\mathcal{L},k}^b = [\mathbf{v}_{kt}^b]_{t=1}^{T_0}$ and $\mathbf{V}_{\mathcal{U},k}^b = [\mathbf{v}_{kt}^b]_{t=T_0+1}^T$; their concatenation, $\mathbf{V}_k = [\mathbf{V}_{\mathcal{L},k}^b \mathbf{V}_{\mathcal{U},k}^b] = [\mathbf{v}_{kt}^b]_{t=1}^T$; and the concatenation over all persons, i.e., the concatenation of all body visual features, $\mathbf{V}^b = [\mathbf{V}_k^b]_{k=1}^K$. The matrices $\Theta_{\mathcal{L},k}^b$, $\Theta_{\mathcal{U},k}^b$, Θ_k^b , and Θ^b are analogously built from θ_{kt}^b .

The objective is to estimate the matrix $\Theta_{\mathcal{U}} = [\Theta_{\mathcal{U},k}^b]_{k=1}^K$, under the assumption of a linear classifier,¹

$$\Theta^b = \mathbf{W}^b \begin{bmatrix} \mathbf{V}^b \\ \mathbf{1}^\top \end{bmatrix}, \quad (3.1)$$

with $\mathbf{W}^b \in \mathbb{R}^{C \times (d_b+1)}$. Remarkably, the joint feature-label matrix $\mathbf{J}_b = \left[\Theta^{b\top} \mathbf{V}^{b\top} \mathbf{1} \right]^\top \in \mathbb{R}^{(C+d_b+1) \times KT}$ is low rank, since the linear classifier in Equation 3.1 imposes a linear dependency between the rows of \mathbf{J}_b . Therefore, we set the following optimization problem for $\Theta_{\mathcal{U}}^b$:

1. We assume that the non-linearity is absorbed by the feature extraction procedure.

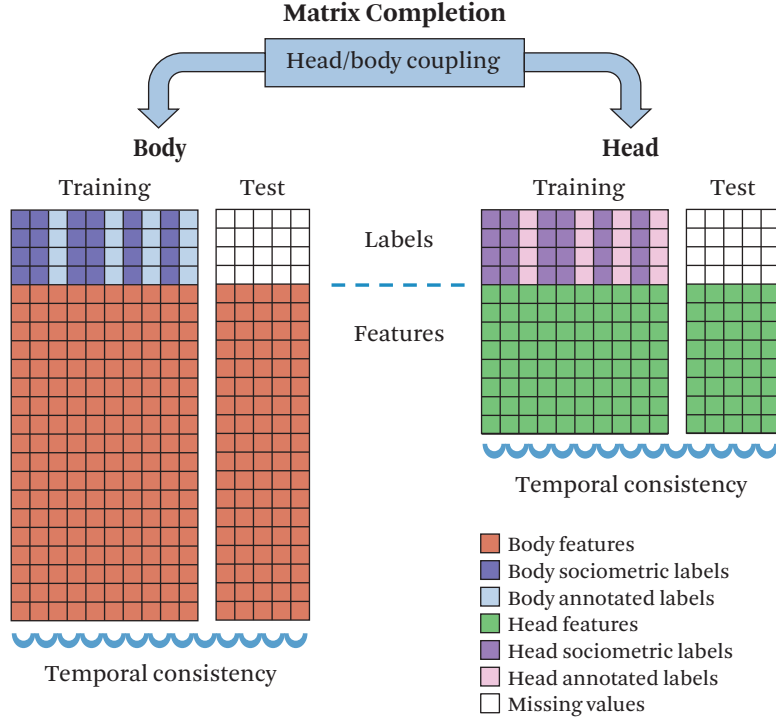


Figure 3.3 Illustration of the proposed matrix completion framework for multimodal head and body pose estimation (MC-HBPE). Two matrix completion problems, for the head and body poses, are regularized to account for the temporal consistency and for head/body coupling. (From Alameda-Pineda et al. [2015])

$$\Theta_{\mathcal{U}}^{b*} = \arg \min_{\Theta_{\mathcal{U}}^b} \text{rank}(\mathbf{J}_b). \quad (3.2)$$

Minimizing the rank of a matrix is an NP-hard problem, but it can be exactly relaxed by means of the nuclear norm [Candès and Tao 2010]. Indeed, since the nuclear norm $\|\cdot\|_*$ is the tightest convex envelope of the rank, the previous optimization problem is equivalent to:

$$\Theta_{\mathcal{U}}^{b*} = \arg \min_{\Theta_{\mathcal{U}}^b} \|\mathbf{J}_b\|_*. \quad (3.3)$$

In real applications, both the observations and the training labels are noisy. Therefore, it is of crucial importance to account for the observational and label noise. In practice, we assume $\tilde{\mathbf{J}}_b = \mathbf{J}_b + \mathbf{E}_b$, where $\tilde{\mathbf{J}}_b$ represents the noisy features and labels, and \mathbf{E}_b represents the noise. The objective is to estimate the low-rank matrix \mathbf{J}_b that best approximates the observed matrix $\tilde{\mathbf{J}}_b$. This is done by

constraining Equation 3.3, and relaxing it into [Goldberg et al. 2010]:

$$\min_{\mathbf{J}_b} \nu_b \|\mathbf{J}_b\|_* + \frac{\lambda_b}{2} \left\| P_{\mathcal{O}}^b (\tilde{\mathbf{J}}_b - \mathbf{J}_b) \right\|_{\mathcal{F}}^2, \quad (3.4)$$

where ν_b and λ_b are regularization parameters, $\|\mathbf{X}\|_{\mathcal{F}}^2 = \text{Tr}(\mathbf{X}^{\top}\mathbf{X})$ denotes the Frobenius norm of \mathbf{X} , and $P_{\mathcal{O}}^b$ denotes the projection onto the “set of observations,” formally $\mathcal{L} \cup \mathcal{U}$, defined as $P_{\mathcal{O}}^b \left(\left[\Theta^{\text{b}\top} \mathbf{V}^{\text{b}\top} \mathbf{1} \right]^{\top} \right) = \left[\left[\Theta_{\mathcal{L},k}^{\text{b}\top} \mathbf{0} \right]_{k=1}^K \mathbf{V}^{\text{b},\top} \mathbf{1} \right]^{\top}$, and thus setting to zero all elements associated to $\Theta_{\mathcal{U}}^b$. We remark that this *unitary* regularization term encompasses the error on the visual features as well as on the training labels.

One of the prominent factors alleviating the matrix completion problem with noisy observations is the inherent temporal structure of the problem. Intuitively, we would expect the labels to be *smooth* in time, that is, $\theta_{kt}^b \approx \theta_{kt+1}^b, \forall k, t$. We reinforce this intuition through a set of *binary* regularization terms, grouped in the following loss function:

$$T_b(\mathbf{J}_b) = \frac{1}{2} \text{Tr} \left(P_{\Theta}^b(\mathbf{J}_b)^{\top} \mathbf{T}_b P_{\Theta}^b(\mathbf{J}_b) \right), \quad (3.5)$$

where P_{Θ}^b is the projection onto the labels, Θ_b , defined as $P_{\Theta}^b \left(\left[\Theta^{\text{b},\top} \mathbf{V}^{\text{b},\top} \mathbf{1} \right]^{\top} \right) = \left[\Theta^{\text{b},\top} \mathbf{0} \mathbf{0} \right]^{\top}$, and $\mathbf{T}_b \in \mathbb{R}^{KT \times KT}$ is the Laplacian matrix associated to the graph encoding the relations between the variables. In our case, $\mathbf{T}_b = \mathbf{I}_K \otimes \mathbf{L}$, where \mathbf{I}_K is the K -dimensional identity matrix, \otimes is the Kronecker product, and $\mathbf{L} \in \mathbb{R}^{T \times T}$ is a tri-diagonal matrix defined as follows: the elements of the two subdiagonals are set to -1 , and the elements of the main diagonal are set to 2 , except $\mathbf{L}_{11} = \mathbf{L}_{TT} = 1$. In practice this means that only contiguous pose labels of the same person are taken to be equal.

Until this point, we have discussed the problem of body and head pose estimation separately. However, previous research studies in head and body pose estimation have demonstrated the synergistic effect of joint estimation [Chen and Odobez 2012]. Subsequently, we will show that the estimation of head and body pose benefits from a coupling structure, in addition to the temporal structure already discussed. We write $\theta_{kt}^b \approx \theta_{kt}^h$, and formalize the coupling structure by means of the following regularization term:

$$C(\mathbf{J}_b, \mathbf{J}_h) = \frac{1}{2} \left\| P_{\Theta}^b(\mathbf{J}_b) - P_{\Theta}^h(\mathbf{J}_h) \right\|_{\mathcal{F}}^2, \quad (3.6)$$

where P_{Θ}^h is defined analogously² as P_{Θ}^b . Notice that the regularization term does not imply that the body and the head orientation are identical; rather, it indicates that the head and body poses belong to the same sector (recall the formal definition of $\theta_{kt}^{d_b}$ and $\theta_{kt}^{d_h}$) most of the time.

Summarizing, the head and body pose estimation is cast into a matrix completion problem encompassing the temporal and coupling structural constraints, by considering the following optimization problem:

$$\begin{aligned} \min_{\mathbf{J}_b, \mathbf{J}_h} & \nu_b \|\mathbf{J}_b\|_* + \frac{\lambda_b}{2} \left\| P_{\Theta}^b (\tilde{\mathbf{J}}_b - \mathbf{J}_b) \right\|_{\mathcal{F}}^2 + \nu_h \|\mathbf{J}_h\|_* + \frac{\lambda_h}{2} \left\| P_{\Theta}^h (\tilde{\mathbf{J}}_h - \mathbf{J}_h) \right\|_{\mathcal{F}}^2 \\ & + \frac{\tau_b}{2} \text{Tr} \left(P_{\Theta}^b (\mathbf{J}_b)^{\top} \mathbf{T}_b P_{\Theta}^b (\mathbf{J}_b) \right) + \frac{\tau_h}{2} \text{Tr} \left(P_{\Theta}^h (\mathbf{J}_h)^{\top} \mathbf{T}_h P_{\Theta}^h (\mathbf{J}_h) \right) \\ & + \frac{\lambda_c}{2} \left\| P_{\Theta}^b (\mathbf{J}_b) - P_{\Theta}^h (\mathbf{J}_h) \right\|_{\mathcal{F}}^2. \end{aligned} \quad (3.7)$$

This is a non-linear convex optimization problem whose critical points cannot be expressed in closed-form solution. In the following we, describe the proposed optimization algorithm to find the optimal solution for Equation 3.7.

3.4.2 Optimization Method

The alternating-direction method of multipliers (ADMM) is a well-known optimization procedure, and a review of theoretical results, practical implementation considerations, and applications can be found in [Boyd et al. \[2011\]](#). Recently, an ADMM-based optimization procedure for the matrix completion problem has been presented [[Kalofolias et al. 2014](#)]. We also considered ADMM to solve the matrix completion problem. However, the proposed method is intrinsically different from previous approaches because of two reasons:

- First, we consider a coupled matrix completion problem, which is different from solving the concatenated head/body matrix. Indeed, completing the (temporal-wise) concatenated matrix implicitly assumes not only that the body and head visual features have the same dimension, but also that the linear classifier for the body and head poses are equal, $\mathbf{W}^b = \mathbf{W}^h$. If we consider the concatenation in the other dimension, we would use the method to find a linear relationship between the head and body features, on top of the two classifiers. This is potentially dangerous since optimizing this relationship would not help to obtain better classifiers. Therefore, the method developed

2. Even if P_{Θ}^b and P_{Θ}^h are conceptually equivalent, they are formally different since the features' dimension is different in general, $d_b \neq d_h$.

in Kalofolias et al. [2014] cannot be used in our case. Instead, we propose and develop a *coupled alternating direction method of multipliers (C-ADMM)*.

- Second, the temporal structure of the problem is modeled with a column-wise regularizer on the matrices, which is a particular case of Kalofolias et al. [2014]. Importantly, as will be shown in the following, column-wise regularization can be solved in closed form, while the formulation of Kalofolias et al. [2014] cannot. Consequently, all the steps of the proposed C-ADMM have a closed-form solution, and the convergence is guaranteed.

Classically, the ADMM arises from the construction of the *augmented Lagrangian* [Boyd et al. 2011], which in our case we write as:

$$\begin{aligned} \mathcal{L} = & \nu_b \|\mathbf{J}_b\|_* + \frac{\lambda_b}{2} \left\| P_{\Theta}^b(\tilde{\mathbf{J}}_b - \mathbf{K}_b) \right\|_{\mathcal{F}}^2 + \nu_h \|\mathbf{J}_h\|_* + \frac{\lambda_h}{2} \left\| P_{\Theta}^h(\tilde{\mathbf{J}}_h - \mathbf{K}_h) \right\|_{\mathcal{F}}^2 \quad (3.8) \\ & + \frac{\tau_b}{2} \text{Tr} \left(P_{\Theta}^b(\mathbf{K}_b)^{\top} \mathbf{T}_b P_{\Theta}^b(\mathbf{K}_b) \right) + \frac{\tau_h}{2} \text{Tr} \left(P_{\Theta}^h(\mathbf{K}_h)^{\top} \mathbf{T}_h P_{\Theta}^h(\mathbf{K}_h) \right) \\ & + \frac{\lambda_c}{2} \left\| P_{\Theta}^b(\mathbf{K}_b) - P_{\Theta}^h(\mathbf{K}_h) \right\|_{\mathcal{F}}^2 + \frac{\phi_b}{2} \|\mathbf{K}_b - \mathbf{J}_b\|_{\mathcal{F}}^2 + \frac{\phi_h}{2} \|\mathbf{K}_h - \mathbf{J}_h\|_{\mathcal{F}}^2 \\ & + \langle \mathbf{M}_b, \mathbf{J}_b - \mathbf{K}_b \rangle + \langle \mathbf{M}_h, \mathbf{J}_h - \mathbf{K}_h \rangle. \end{aligned}$$

The Lagrangian \mathcal{L} has to be minimized with respect to $\mathbf{J}_b, \mathbf{J}_h, \mathbf{K}_b, \mathbf{K}_h, \mathbf{M}_b$, and \mathbf{M}_h , where the last two are matrices of Lagrangian multipliers, \mathbf{K}_b and \mathbf{K}_h are auxiliary variables, and $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{ij} \mathbf{A}_{ij} \mathbf{B}_{ij}$ is the scalar product in the matrix space.

At iteration $r + 1$, the C-ADMM updates are:

$$\left(\mathbf{J}_b^{r+1}, \mathbf{J}_h^{r+1} \right) = \arg \min_{\mathbf{J}_b, \mathbf{J}_h} \mathcal{L} \left(\mathbf{J}_b, \mathbf{J}_h, \mathbf{K}_b^r, \mathbf{K}_h^r, \mathbf{M}_b^r, \mathbf{M}_h^r \right), \quad (3.9)$$

$$\left(\mathbf{K}_b^{r+1}, \mathbf{K}_h^{r+1} \right) = \arg \min_{\mathbf{K}_b, \mathbf{K}_h} \mathcal{L} \left(\mathbf{J}_b^{r+1}, \mathbf{J}_h^{r+1}, \mathbf{K}_b, \mathbf{K}_h, \mathbf{M}_b^r, \mathbf{M}_h^r \right), \quad (3.10)$$

$$\mathbf{M}_b^{r+1} = \mathbf{M}_b^r + \phi_b \left(\mathbf{J}_b^{r+1} - \mathbf{K}_b^{r+1} \right), \quad (3.11)$$

$$\mathbf{M}_h^{r+1} = \mathbf{M}_h^r + \phi_h \left(\mathbf{J}_h^{r+1} - \mathbf{K}_h^{r+1} \right). \quad (3.12)$$

From Equation 3.8 it is straightforward, that the coupling term does not have any effect on Equation 3.9, but only on Equation 3.10. Therefore, solving for Equation 3.9, the two matrices $\mathbf{J}_b^{r+1}, \mathbf{J}_h^{r+1}$ can be computed independently. Furthermore, the optimal solution of Equation 3.9 is:

$$\mathbf{J}_b^{r+1} = \mathbf{U}_b S_{\frac{\nu_b}{\phi_b}} \left(\mathbf{D}_b \right) \mathbf{V}_b, \quad (3.13)$$

where $\mathbf{U}_b \mathbf{D}_b \mathbf{V}_b$ is the singular-value decomposition of the matrix $\mathbf{K}_b^r - \frac{1}{\phi_b} \mathbf{M}_b^r$ and $S_\lambda(x) = \max(x - \lambda, 0)$ is the shrinkage operator with constant λ applied element-wise to the diagonal matrix of singular values, \mathbf{D}_b . \mathbf{J}_h^{r+1} is found analogously. The details of the derivation can be found in the supplementary material.

The second C-ADMM update, i.e., Equation 3.10, is also solved in closed form. The critical point is computed by canceling the derivative of the objective function in Equation 3.10 with respect to \mathbf{K}_b and \mathbf{K}_h . As shown in the supplementary material, these derivatives lead to block-diagonal linear systems that can be efficiently solved. In order to sketch the proof, we define $\mathbf{k}_b = \text{vec}(\mathbf{K}_b)$ as the row vectorization of \mathbf{K}_b . The same notation stands for all other matrices involved. Moreover, we define $\mathbf{k}_{b,kc} = [(\theta_{kt}^b)_c]_{t=1}^T$ as the T -dimensional vector composed of the coordinate c of the angle labels of person k over time. Likewise, $\mathbf{k}_{b,v}$ stands for the $KT(d_b + 1)$ -dimensional row vectorization of the rest of the matrix \mathbf{K}_b , that is, the part corresponding to the visual features \mathbf{V}^b . With these notations, the vector \mathbf{k}_b is rewritten as:

$$\mathbf{k}_b = \left[\mathbf{k}_{b,11}^\top \cdots \mathbf{k}_{b,KC}^\top \mathbf{k}_{b,v}^\top \right]^\top,$$

and analogously for all other row-vectorized matrices.

Each of the KC variables for body and head, $\mathbf{k}_{b,kc}$ and $\mathbf{k}_{h,kc}$, are independently solved:

$$\mathbf{k}_{b,kc} = (\mathbf{L}_h \mathbf{L}_b - \mathbf{I}_T)^{-1} (\mathbf{L}_h \bar{\mathbf{k}}_{b,kc} + \bar{\mathbf{k}}_{h,kc}), \quad (3.14)$$

$$\mathbf{k}_{h,kc} = \mathbf{L}_b \mathbf{k}_{b,kc} - \bar{\mathbf{k}}_{h,kc}, \quad (3.15)$$

where all matrices and vectors are precisely defined in the supplementary material. The vector $\bar{\mathbf{k}}_{b,kc}$ is computed from $\tilde{\mathbf{j}}_{b,kc}$, $\mathbf{m}_{b,kc}^r$, and $\mathbf{j}_{b,kc}^{r+1}$ and thus depends on the iteration (analogously for the head). Importantly, the matrices \mathbf{L}_b and \mathbf{L}_h , computed from \mathbf{L} and the regularization parameters, do not depend on the iteration. Therefore, the inversion can be computed before the iterative procedure of the C-ADMM starts, leading to an efficient algorithm. The system involving $\mathbf{k}_{b,v}$ and $\mathbf{k}_{h,v}$ is simpler since there is no coupling between body and head features and there is no temporal regularization:

$$(\lambda_b + \phi_b) \mathbf{k}_{b,v} = \lambda_b \tilde{\mathbf{j}}_{b,v} + \mathbf{m}_{b,v}^r + \phi_b \mathbf{j}_{b,v}^{r+1} \quad (3.16)$$

$$(\lambda_h + \phi_h) \mathbf{k}_{h,v} = \lambda_h \tilde{\mathbf{j}}_{h,v} + \mathbf{m}_{h,v}^r + \phi_h \mathbf{j}_{h,v}^{r+1}. \quad (3.17)$$

Equations 3.14–3.17 provide the critical point sought in Equation 3.10 and conclude the derivation of the proposed C-ADMM. As is classically the case, the complexity bottleneck of the matrix-completion solver is in the computation of the

Algorithm 3.1 The coupled alternating-directions method of multipliers solving for the derived matrix completion for multimodal HBPE

Input:

Observation $\tilde{\mathbf{J}}_h, \tilde{\mathbf{J}}_b$ and Laplacian $\mathbf{T}_h, \mathbf{T}_b$ matrices. Regularization parameters.

Output:

Optimized $\mathbf{J}_h, \mathbf{J}_b$.

Randomly initialize $\mathbf{M}_b, \mathbf{M}_h, \mathbf{K}_b, \mathbf{K}_h$

repeat

Solve for $\mathbf{J}_h, \mathbf{J}_b$ in Equation 3.9 using Equation 3.13

Solve for $\mathbf{K}_h, \mathbf{K}_b$ in Equation 3.10 using Equation 3.14–Equation 3.17

Solve for $\mathbf{M}_h, \mathbf{M}_b$ using Equation 3.11–Equation 3.12

until convergence

singular-value decomposition required in Equation 3.13. The complete C-ADMM algorithm is shown in Algorithm 3.1.

3.5 Experiments

In this section, we show the results of our experimental evaluation conducted on the novel SALSA dataset [Alameda-Pineda et al. 2016b]. Then, we show the results on estimating the head and body orientations with the matrix-completion framework. We also perform further experiments on the analysis of social interaction in SALSA, showing that the computed head and body orientations are accurate and robust so as to efficiently detect F-formations. Overall, this demonstrates the potential of multimodal approaches for the analysis of social interactions.

3.5.1 Head and Body Pose Estimation

To experimentally validate the technical core of the present study, i.e., the MC-HBPE, we consider those frames of the poster session of SALSA in which all the participants are in the field of view of the four cameras. In practice, we use 340 frames of the poster session (17 minutes approximately), resulting in a dataset comprising 6120 head/body samples. We split this set into two equal parts, and use only the annotations of the first part for training: 10% of manual annotations and 90% of weak annotations obtained with the sociometric badges, when available. In essence, we have manual annotations for head and body pose in 17 frames, and therefore the labels are sparse. Overall, this is quite a realistic setup since we only necessitate annotations for less than 20 frames for the method to work. This feature arises directly from the transductive nature of matrix completion. We report

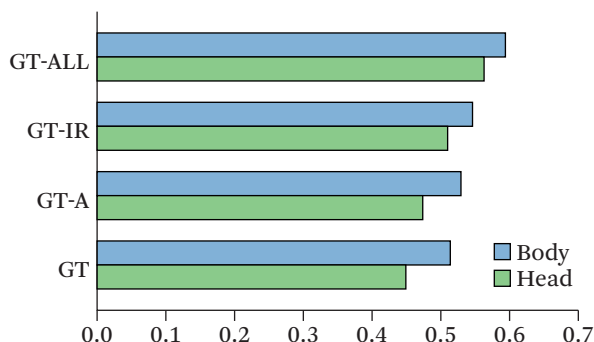


Figure 3.4 Accuracy of the head and body pose estimates obtained with the proposed MC-HBPE approach using visual features and labels from different modalities: ground truth annotations (GT), GT and audio data (GT-A), GT and infrared data (GT-IR), all modalities (ALL).

the classification accuracy of the test samples, i.e., the second half of the dataset. In all our experiments, and for both our approach and the baseline methods, the regularization parameters are set with cross-validation.

Figure 3.4 shows the performance obtained with our method when the training labels come from different modalities. In light of the results, we can clearly state that using multimodal labels significantly improves the accuracy of both head and body pose estimates. Interestingly, when considering only ground truth and auditory labels (i.e., no additional labels for body pose), not only the accuracy of the head pose estimates improves, but also the accuracy of body pose estimates. This is an important and remarkable effect of the head/body coupling term introduced in the proposed formulation. We notice the analogous effect with the performance rise not only of body pose estimates, but also head pose estimates, when adding the infrared labels (i.e., with no additional labels for head pose). It is worth noticing that infrared labels have a significantly more positive effect than auditory labels. We ascribe this fact to the higher accuracy of the infrared labels (87%) compared to the auditory labels (70%) when compared to the manual annotations. Importantly, we highlight the synergistic effect of jointly using auditory, infrared, and visual data to robustly solve the head and body pose estimation problem by means of matrix completion techniques. Finally, the implementation of the proposed methodology needs roughly 0.5 seconds to estimate the head and body poses of one person.

Table 3.2 reports the classification accuracy of several methods. First of all, we detail the proposed MC-HBPE method considering different configurations regarding the head/body coupling and the temporal consistency (from top to bottom):

Table 3.2 Classification accuracy of different methods for HBPE

Method	C	T_b, T_h	Body	Head
MC-HBPE	✓		0.515	0.465
		✓	0.548	0.515
	✓	✓	0.597	0.567
Chen and Odobez [2012]			0.524	0.475
TSVM [Joachims 1999]			0.481	0.429
Goldberg et al. [2010]			0.479	0.439

(i) only with head-body coupling $\tau_b, \tau_h = 0$ in Equation 3.7; (ii) only with temporal consistency λ_c equal to 0 in Equation 3.7; and (iii) with both types of constraints. The comparison with the state of the art is particularly difficult in our case, since there is no existing methodology for head and body pose estimation using multimodal data. That is why we chose to compare it with the state of the art on *visual-only* HBPE [Chen and Odobez 2012]. Notably, the conversational flavor of SALSA is challenging for Chen and Odobez [2012], since when analyzing FCGs the direction of motion cannot be considered as a proxy for the body pose, as done in Chen and Odobez [2012]. Moreover, given that the proposed learning methodology belongs to the class of transductive learning approaches, and Chen and Odobez [2012] does not, we perform additional comparisons with transductive support vector machines (TSVM) [Joachims 1999]. It is worth noting that neglecting the coupling and the temporal consistency terms in MC-HBPE, i.e., $\tau_b, \lambda_c, \tau_h = 0$ in Equation 3.7, is similar to the work of Goldberg et al. [2010]. Therefore, MC-HBPE could be seen as a generalization of the transductive methodology presented in Goldberg et al. [2010]. Results in Table 3.2 demonstrate the advantageous performance obtained when enforcing temporal consistency or coupling the head and body pose estimates. Importantly, the effect of both regularizations is impressive, increasing the classifiers' accuracy by approximately 20%. Our understanding is that this is the main reason for which the complete method we presented outperforms the baselines, up to a large extent. In order to show the complexity of the task, we show the original image of one of the cameras and the estimated poses plot in a bird's-eye view in Figure 3.5. Finally, we evaluate the robustness of the approach to the amount of training and of manually annotated data in Figure 3.6, where we note the 40% accuracy obtained with only 1% of manually annotated data (plus the noisy labels from the sociometric badges).



Figure 3.5 Frame example: (left) the original image of one of the cameras, (right) the head–green and body–red pose estimates plot in bird’s-eye view.

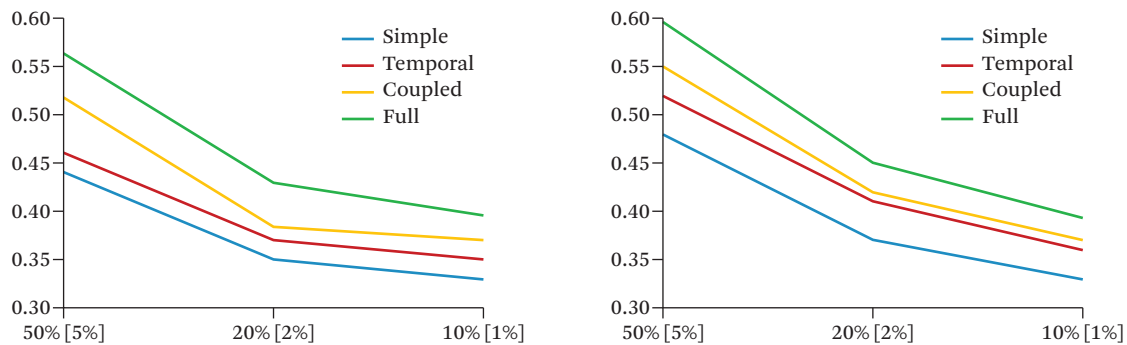


Figure 3.6 Classification accuracy (left: head pose, right: body pose) for different percentages of training samples and of manually annotated samples (in square brackets).

3.5.2 F-formation Detection

One of the prominent applications of robust estimation of the head and body poses when analyzing FCGs is the detection of F-formations. More precisely, we aim to detect the different conversational groups and who belongs to each group. We evaluate three state-of-the-art approaches for F-formation detection, namely: the Hough voting method (HVFF-lin) in [Cristani et al. \[2011\]](#), its multi-scale extension (HVFF-ms) [[Setti et al. 2013](#)], and the graph-cut approach in [Setti et al. \[2015\]](#). We chose these methods because their implementation is publicly available.³ All these approaches compute the F-formation independently frame by frame, from the targets’ position and pose. The rationale behind the Hough voting methods is

3. <http://profs.sci.univr.it/~cristanm/ssp/>

that each target generates a set of samples in front of him/her, which are candidates for the F-formation centers. By aggregating these samples into a discrete number of cells, F-formations are detected by finding local maxima in the discretized space. Conversely, in the graph-cut algorithm [Setti et al. 2015], an optimization problem is solved, discovering the coordinates of the F-formations directly.

Previous work [Hung and Kröse 2011] has acknowledged the importance of accurate body pose estimates for detecting F-formations. However, one of the inherent difficulties within the analysis of FCGs in crowded indoor environments is the adverse impact of highly occluded body images, which exacerbate strong noise and outliers in the estimated body poses. Classically, the head pose is used as a surrogate for the body pose, and the F-formations are detected from the head pose. In Section 3.5.1, we demonstrate that the proposed MC-HBPE approach can be effectively used to accurately infer the body pose, thus making it possible to use the body pose for F-formation detection and therefore improving the quality of the analysis of FCGs.

We implement the validation protocol defined in Cristani et al. [2011], and report F-formation precision, recall, and F1-measure in Table 3.3. At each frame, we consider a group as correctly estimated if at least $\rho \cdot |G|$ of the members are correctly found, and if no more than $1 - \rho \cdot |G|$ non-members are wrongly identified, where $|G|$ is the cardinality of the group G , and $\rho = 2/3$. Table 3.3 shows the performance measures using body (first three rows) and head (fourth and fifth rows) pose. We compare the estimates obtained with MC-HBPE to the body pose ground truth and the body estimates computed by Chen and Odobez [2012]. Regarding the head pose, in addition to our approach, we also use the TSVM approach in Joachims [1999] to emulate previous work on F-formation detection that used head pose estimates. The results clearly confirm that accurate inference of the body pose using the MC-HBPE framework is advantageous for the detection of F-formations.

3.5.3 Comparison with Joint F-formation and Body Pose Estimation

In this section we compare the proposed approach with a recent method for joint F-formation detection and body pose estimation, which also operates in a transductive setting [Ricci et al. 2015]. The main difference between the proposed framework and the method in Ricci et al. [2015] is that a different type of weak supervision is exploited for inferring the head and body pose of the targets. While in this work we consider information derived from wearable sensors to obtain noisy labels for learning the head and body pose classifiers, in Ricci et al. [2015] the spatial configuration of conversational groups is exploited. Indeed, for unlabeled samples it is reasonable to constrain the head and body pose of a target to be consistent with

Table 3.3 Performance of state-of-the-art approaches on F-formation detection from different estimations of the head and body pose

	Cristani et al. [2011]	Setti et al. [2013]	Setti et al. [2015]
Our-B	0.62 / 0.61 / 0.62	0.66 / 0.64 / 0.66	0.69 / 0.66 / 0.67
Prev.-B	0.56 / 0.58 / 0.58	0.59 / 0.62 / 0.61	0.62 / 0.61 / 0.62
GT-B	0.66 / 0.65 / 0.65	0.72 / 0.69 / 0.71	0.74 / 0.72 / 0.73
Our-H	0.61 / 0.59 / 0.60	0.64 / 0.63 / 0.63	0.65 / 0.64 / 0.64
Prev.-H	0.58 / 0.58 / 0.58	0.59 / 0.58 / 0.59	0.62 / 0.59 / 0.61
GT-H	0.65 / 0.63 / 0.64	0.70 / 0.67 / 0.69	0.72 / 0.70 / 0.71

“Our” stands for ours.

“Prev.” stands for body in [Chen and Odobez \[2012\]](#) and for head in [Joachims \[1999\]](#), respectively.

“GT” stands for ground truth.

Each table cell contains precision, recall, and F1-measure, in that order.

the position of the center of the F-formation the target belongs to. In the following, we provide a brief description of the method in it and [Ricci et al. \[2015\]](#) and an experimental comparison between the proposed framework.

Similarly to the proposed method, in [Ricci et al. \[2015\]](#) the multi-target tracking algorithm in [Lanz \[2006\]](#) is applied to estimate the targets’ positions and the head localization method in [Yan et al. \[2013\]](#) is used to derive head and body crops. Then, HOG features are extracted from head and body images. These features and the targets’ positions are given as input to a learning model to infer the head and body pose of all the targets in the social scene and simultaneously detect conversational groups.

More formally, given a video depicting K people involved in a social gathering, the head and body bounding boxes are extracted for each target k at each frame t . In other words, for each target k a set of samples $\mathcal{S}_k = \{\mathbf{x}_{k,t}^b, \mathbf{x}_{k,t}^h\}_{t=1}^T$ is obtained, where $\mathbf{x}_{k,t}^b \in \mathbb{R}^{d_b}$, $\mathbf{x}_{k,t}^h \in \mathbb{R}^{d_h}$ are the HOG descriptors associated with the head and body bounding boxes and T denotes the number of frames in the video. Together with unlabeled samples of the social scene, some annotated training samples derived from an auxiliary dataset are considered, i.e., $\mathcal{T}^b = \{(\hat{\mathbf{x}}_i^b, \mathbf{y}_i^B)\}_{i=1}^{N_b}$, $\mathcal{T}^h = \{(\hat{\mathbf{x}}_i^h, \mathbf{y}_i^H)\}_{i=1}^{N_h}$, where $\hat{\mathbf{x}}_i^b \in \mathbb{R}^{d_b}$, $\hat{\mathbf{x}}_i^h \in \mathbb{R}^{d_h}$ are HOG features extracted from body and head crops, respectively; $\mathbf{y}_i^B \in \{0, 1\}^{N_C^b}$, $\mathbf{y}_i^H \in \{0, 1\}^{N_C^h}$ are the corresponding pose labels. The set of possible head and pose directions is quantized into $N_C^h = N_C^b = 8$ possible classes.

The learning algorithm presented in Ricci et al. [2015] is designed to simultaneously learn two classifiers, one for head pose f^H and the other for body pose f^B , and detect F-formations by clustering targets belonging to the same group (i.e., finding the matrix $\mathbf{C} = [\mathbf{c}_{1,1}, \dots, \mathbf{c}_{N_K, N_T}]$ of the centers $\mathbf{c}_{k,t}$ of the conversational groups). The following optimization problem is proposed:

$$\min \mathcal{L}(f^H, f^B, \mathbf{C}) = \mathcal{L}^H(f^H) + \mathcal{L}^B(f^B) + \mathcal{C}(f^H, f^B) + \mathcal{F}(f^B, \mathbf{C}). \quad (3.18)$$

The first three terms in \mathcal{L} leverage information from both annotated and unlabeled samples in order to learn both the head and body pose classifiers in a coupled fashion. Specifically, the first two terms, corresponding respectively to head and body samples, are defined by combining a loss function on labeled data with a graph-based regularization term integrating information about unlabeled data, as typically done in semi-supervised learning methods [Zhu and Goldberg 2009], i.e.:

$$\mathcal{L}^Z = \sum_{n=1}^{N^Z} \|f^Z(\hat{\mathbf{x}}_n^Z) - (\mathbf{y}_n^Z)\|_{\mathbf{M}}^2 + \lambda_R \|f^Z\|^2 + \lambda_U \sum_{i,j} \omega_{ij} \|f^Z(\mathbf{x}_{i,t}^Z) - f^Z(\mathbf{x}_{j,t}^Z)\|_{\mathbf{M}}^2, \quad (3.19)$$

where $Z = \{B, H\}$ and ω_{ij} are user-defined parameters designed to take into account unlabeled samples' similarity. The matrix \mathbf{M} models the mapping from the pose label vectors to angles and $\|\mathbf{g}\|_{\mathbf{M}} = \sqrt{\mathbf{g}^T \mathbf{M} \mathbf{g}}$.

The term \mathcal{C} is introduced to guarantee coherence between head and body pose estimates on each target such as to reflect human anatomical constraints, i.e.:

$$\mathcal{C} = \sum_{k=1}^K \sum_{t=1}^T \|f^B(\mathbf{x}_{k,t}^B) - f^H(\mathbf{x}_{k,t}^H)\|_{\mathbf{M}}^2. \quad (3.20)$$

The term \mathcal{F} models the relationship between body pose and conversational groups. The intuition is that, knowing the body orientation of the targets, F-formations can be detected. Similarly, the body orientation can be estimated more precisely if the centers of the conversational groups are known. The function \mathcal{F} is defined as follows:

$$\mathcal{F} = \lambda_F \sum_{t=1}^{N_T} \sum_{k=1}^{N_K} \|(\mathbf{p}_{k,t} + D\mathbf{A}f^B(\mathbf{x}_{k,t}^B)) - \mathbf{c}_{k,t}\|^2 + \gamma_C \sum_{t=1}^{N_T} \sum_{k,q=1}^{N_K} \|\mathbf{c}_{k,t} - \mathbf{c}_{q,t}\|_1, \quad (3.21)$$

where $\mathbf{A} = [\cos \alpha_1, \dots, \cos \alpha_{N_C}; \sin \alpha_1, \dots, \sin \alpha_{N_C}]$, α_j are the angles corresponding to the different body pose classes, and D is a user-defined parameter indicating the distance of a target from an F-formation center.

The function \mathcal{F} must be minimized in order to learn the parameters \mathbf{C} , corresponding to the centers of the conversational groups, and the body classifier f^B . In-

Table 3.4 Comparison between the proposed framework and Ricci et al. [2015]

Method	SALSA		
	Head Pose Error	Body Pose Error	F-Formation F1 Measure
Our approach	49.8°	51.6°	0.67
Ricci et al. [2015]	50.7°	50.2°	0.67

tuitively, on the one hand, if the body classifier is given, the term $\mathbf{p}_{k,t} + D\mathbf{A}f^B(\mathbf{x}_{k,t}^B)$ generates a set of possible center locations associated with predicted targets' body orientation. Then, by minimizing Equation 3.21, these locations are clustered, finding the F-formation centers $\mathbf{c}_{k,t}$. On the other hand, if the centroids $\mathbf{c}_{k,t}$ are given, minimizing Equation 3.21 implies constraining the body pose classifier to output a target pose that is more toward the center of the conversational group. The parameter γ_c regulates the number of groups detected. In practice, singleton groups are obtained for small values of γ_c and all targets are merged in a single group as $\gamma_c \rightarrow \infty$. The optimization problem in Equation 3.18 is solved with an alternate optimization method. The reader is referred to Ricci et al. [2015] for details on the optimization algorithm.

We now compare the proposed framework and the method in Ricci et al. [2015], showing results of an experimental evaluation on the SALSA dataset. We analyze performance of the two methods both for head and body pose estimation and for F-formation detection. Table 3.4 compares the performance of the two methods on the three different tasks. As shown in the table, the two approaches achieve comparable performance when applied to the same data, confirming the fact that both sources of weak supervision (i.e., wearable sensors data and groups information) are beneficial. Indeed, the two approaches are somehow complementary, and we believe that further improvement in performance can be achieved by devising a strategy to combine them.

3.6 Conclusion

In this chapter we described recent work on the analysis of free-standing conversational groups using multimodal data gathered from a fixed-camera network and wearable sensors. We discussed a transductive approach for joint estimation of head and body pose which successfully exploits infrared and audio signals collected with sociometric badges to derive head and body labels. To handle noisy features and labels, we devised a coupled matrix completion framework, which

also accounts for the temporal consistency and the head/body pose coupling due to human anatomical constraints. Methodologically speaking, matrix completion is not the only possible choice, but it has three prominent advantages with regard to the experimental setup and the applicative scenario. First of all, the final learning algorithm is alternatively solving convex optimization problems, which makes it intrinsically efficient and therefore attractive for online applications and embedded systems. Second, matrix completion is transductive by nature, thus able to satisfactorily exploit unlabeled data to regularize the classifier. Third, we gain the capacity to model missing data, leading to a flexible framework able to use all (partially and sparsely) available labels and features.

Our study opens the door to several interesting future research directions. First of all, our approach can be extended to take into account conversational groups during the learning phase. Indeed, knowledge about which persons are involved in the same conversation (discussing together) is great prior information for the estimation of the head and body poses. Second, a holistic framework to jointly track people positions and head and body pose would probably be more effective than addressing each single task independently. Finally, the spectrum of applications of the proposed method should be enlarged to address other high-level tasks, such as detecting the addressee and the addresser. In general, this line of research could be useful to endow a robotic platform with the ability to understand the current communicative situation and to automatically shape its behavior, thus adapting to the environment. Similarly, it could be used for surveillance applications and in particular to characterize the social behavior of the individuals in a mall or a train station, although the use of wearable sensors in this applicative scenario would be more restricted. Finally, such automatic analyses could be helpful in extracting behavioral patterns of different subjects of interest (psychiatric patients, children, or elderly people) for different applications (for instance, understanding their social relational patterns, education, and health status monitoring, respectively).

Encrypted Domain Multimedia Content Analysis

Pradeep K. Atrey (University at Albany, SUNY),
Ankita Lathey (University of Winnipeg),
Abukari M. Yakubu (University of Winnipeg)

4.1 Introduction

Multimedia systems produce and transmit a huge amount of content (image, video, and audio) that needs to be processed (real-time) in order to infer its meaning. Also, the storage, distribution, and retrieval of multimedia data is becoming an important task. Irrespective of whether the content processing is performed in real time or in an offline manner, high-end computing infrastructure is required, which is usually very expensive. Today, it is a common practice that such high-end computing tasks are outsourced to a third-party server such as a cloud data center. Such solutions deliver highly scalable and virtualized computing/storage/network resources to efficiently perform the required services.

Since the third-party service providers can often be untrustworthy, their use raises obvious security and privacy concerns. With the invention of modern multimedia processing tools and techniques, it is easy to interpret the information contained in multimedia and mine the content using computer algorithms to gain the usable information such as an individual's identity, current location, movements, and time stamps related to various events [Saini et al. 2012], causing serious damage to the individual's privacy. In order to employ the advantages of outsourcing (i.e., a high-end computing facility) and to overcome its shortcomings (i.e.,

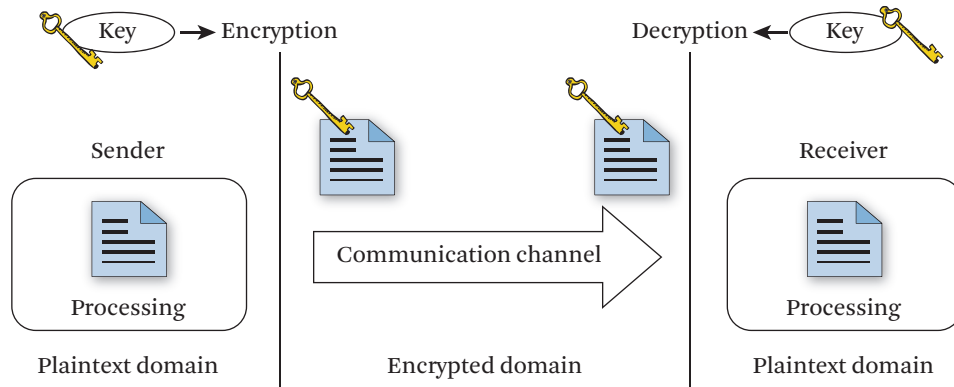


Figure 4.1 Signal processing in plaintext domain.

privacy risks), there is a need for a mechanism to secure the original multimedia content before providing it to the third-party servers for processing, in such a way that an adversary present at these untrusted servers cannot infer anything from the data, but the data can still be processed for the required tasks.

In practice, the most common way to secure multimedia content is to apply cryptographic tools/techniques such as encryption on the original content. However, in order to process the data, a great amount of effort is required at the receiver's end, first to decrypt the data and then to extract its features to operate (see Figure 4.1).

In many real-world scenarios, it is preferable to process the multimedia content directly in an encrypted form. Data processing in encrypted domain requires an operation to provide the same result when it is performed on the encrypted content and when it is applied to the original content [Bianchi et al. 2009b]. The real challenge in the design and development of such a system, as depicted in Figure 4.2, lies in the fact that the content transformation must be done in such a way that the accuracy of the content processing algorithms is not affected, and so that an adversary disguised as a third party is unable to infer any confidential information from the available content. In other words, the result of processing the signal in encrypted domain must be equivalent to the result of processing the same signal in plaintext domain.

Signal processing in encrypted domain (SPED) has been an important area of research for decades. In December 2006, a three-year-long project dedicated to this topic, called SPEED [Erkin et al. 2006], started in Europe. This proved to be a major stepping stone in this area of research. In 2007, there was a special issue published

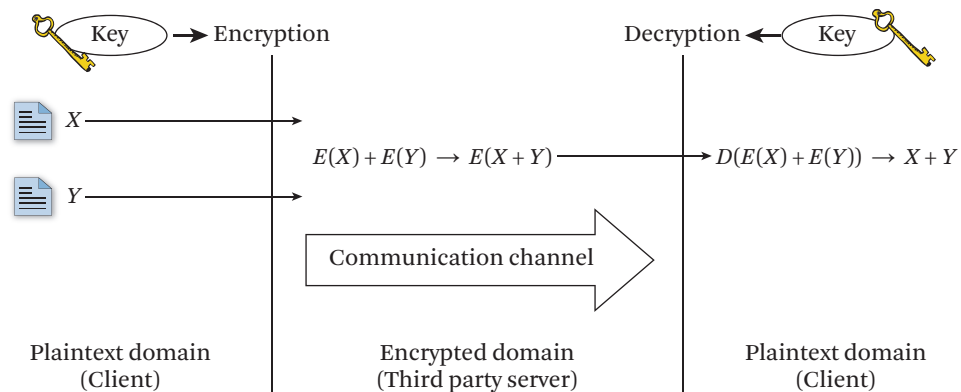


Figure 4.2 Signal processing in encrypted domain articles.

in *EURASIP Journal on Information Security* on SPED [Piva and Katzenbeisser 2008]. Thereafter, many special sessions on SPEED have been organized at various workshops [Bianchi et al. 2009b, Bianchi et al. 2010] and conferences [Barni and Piva 2008, Rane and Barni 2011]. Some papers [Lu et al. 2011, Puech et al. 2012], including a keynote speech, were also presented at various platforms. There are some existing surveys [Erkin et al. 2007, Piva and Katzenbeisser 2008, Prins et al. 2006] that summarize the details of applying cryptographic primitives to signal processing operations in encrypted domain. They consider various signal processing tasks as independent piece-wise operations and present an overview of the homomorphic mathematical details making SPED plausible. There are some works [Fontaine and Galand 2007, Gentry 2009, Legendijk et al. 2013, Aguilar et al. 2013] that present a literature review of the availability of the latest trends and techniques in homomorphic cryptosystems. Such works emphasize the provable security and privacy of the discussed cryptographic primitives, along with their ability to be utilized for SPED and other cloud-based secure processing frameworks.

This chapter emphasizes the need for secure multimedia processing, and reviews the available state-of-the-art literature. We present an overview of the available methods and techniques for processing different kinds of multimedia content (mainly image, video, and audio) in encrypted domain. Past work has been analyzed from different perspectives such as the content processing task performed, the cryptographic technique used, and data and computation overheads incurred. Finally, various open research issues involved with encrypted domain multimedia processing are discussed. It is assumed that readers have a basic understanding of

Table 4.1 A list of the symbols used in the chapter

Symbol Used	Description
$E(\cdot), D(\cdot)$	Encryption and decryption functions, respectively.
\mathcal{Z}_n	A cyclic group or finite field with group of integers $[0, 1, 2, \dots, n - 1]$ under modulo n , n denotes the order or size of the group.
$c(p, r) = g^p \cdot h^r \bmod N$	Commitment value of a random binding factor p ; where g is an element of higher order in \mathcal{Z}_n , h is an element of higher order generated by g and discrete log of g in base h & h is base g must be unknown.

the traditional cryptographic algorithms (e.g., RSA/AES/DDH, etc.) or other mathematical signal processing operations (e.g., SIFT/DCT/FFT/CNF, etc.). To guide the reader, we have provided in Table 4.1 a list of symbols used in this chapter.

The rest of this chapter is organized as follows. In Section 4.2, we present an overview of SPED and discuss its various applications and benefits. Section 4.3 discusses the literature on secure processing of images in encrypted domain. A chronological description of seminal contributions in the domain of processing video data for preserving the privacy of ROI, followed by data processing over encrypted databases in secure domain, are summarized in Section 4.4. Section 4.5 presents a similar overview of progressive realization of SPED techniques used for audio signals. Finally, Section 4.6 concludes the chapter by pointing out some open issues and possible avenues of further research in the area of encrypted domain multimedia processing. A ready reckoner is also provided as a starting point for the naive researchers in this area.

4.2 SPED: An Overview

4.2.1 Background

One of the earliest papers published by [Ahituv et al. 1987] put forward the concept of processing data in encrypted domain. For the first time, the world was presented with the challenges of processing encrypted signals, thereby present-

ing a plethora of opportunities for research in this unexplored area. The authors presented the banking transaction-centric, stepwise performance of arithmetic operations on the encrypted data by taking the following four scenarios into consideration: 1) adding an encrypted data element to plaintext, 2) adding encrypted data to other encrypted data, 3) examining a homomorphic approach for the system to overcome the problem of one-time breaking of the key, and 4) adding an acceptable level of encryption strength. This seminal work presented a detailed description of requirements and algorithms to process the encrypted data, and limitations, for each of the above four cases. It also suggested some future research directions such as addressing more specific practical difficulties in processing encrypted data, actual implementation of the proposed algorithms for typical I/O sequences, interface between the encrypted domain processing, and public-key cryptosystems.

Bianchi et al. [2008a, 2009b], and Erkin et al. [2007] have emphasized that the ability to manipulate signals in encrypted domain is largely based on the following two assumptions:

Homomorphic encryption. Homomorphism is a transformation from one type of algebraic structure into another such that the structure is preserved. This means that for every manipulation to the original data, there is a corresponding manipulation to the transformed data. Furthermore, these manipulations can be classified as additive and multiplicative homomorphism. For example, $D(E(m_1) + E(m_2) \bmod O) = (m_1 + m_2) \bmod O$ and $D(E(m_1) \times E(m_2) \bmod O) = (m_1 \times m_2) \bmod O$ are examples of additive and multiplicative homomorphism, respectively.

Many public-key cryptosystems utilize particular homomorphic properties to carry out the processing of the signals in encrypted domain. They are mostly based on the difficulty of solving some computationally hard and complex problems, such as the ElGamal cryptosystem [Elgamal 1985]—discrete logarithm in finite field with large (prime) numbers, the RSA cryptosystem [Rivest et al. 1978]—factorization of large composite numbers, and the Paillier cryptosystem [Paillier 1999]—deciding if a number is an n^{th} power of \mathcal{Z}_N for a large enough composite N . A table identifying homomorphism in several cryptosystems can be found in Erkin et al. [2007].

Independent sample-wise processing. Encryption is independently applied on individual signal samples. Erkin et al. [2007] state that, although there are no measures applied to hide the temporal or spatial characteristics of the

signal, the use of sophisticated encryption schemes that are semantically secure achieves this property automatically.

Bianchi et al. [2008a, 2009b], and Erkin et al. [2007] have realized that public-key cryptosystems operate on very large algebraic structures. This further facilitated the use of a *probabilistic cryptosystem* [Goldwasser and Micali 1984] instead of a deterministic cryptosystem, to ensure that for any two encrypted signals it is almost impossible to determine if they hide the same sample signal value. The importance of using a probabilistic cryptosystem for encryption lies in the fact that the encryption function is no longer one to one, but one to many, and the decryption function is many to one. There is a huge expansion in the ciphertext space, with its size determined by a random blinding factor (unknown to the decryption function) and the security parameters used. The probabilistic cryptosystems also retain their homomorphic property within the original ciphertext space. The chosen plaintext attack involves listing all possible plaintexts and their corresponding ciphertexts, which becomes computationally hard for a sufficiently large size of the blinding space.

As a solution to data expansion, Bianchi et al. [2008a, 2009b], have proposed to represent an encrypted signal as a reduced-sized, composite signal, allowing linear operations in encrypted domain to speed up via parallel processing. Some fundamental signal processing operations have been implemented: linear filtering [Bianchi et al. 2008a, 2009a], sum-product of two signals [Bianchi et al. 2009b], discrete Fourier transformation [Bianchi et al. 2008b, 2008c], and so on. Their papers discuss application-oriented, case-specific studies in which their methods can be utilized, and the advantages or limitations of having such a system.

Most work on signal processing in encrypted domain is based on client-server architecture where the client is constrained in resources and needs to offload storage and/or computation to the server. These servers can be untrustworthy because they are operated by third-parties outside the firewall of the client. A good example of such architecture is cloud data center (CDC). When designing security models for such scenarios, it is important to take into account the possible behavior of parties involved in the protocol. There are two common attacker models used to categorize such behaviors:

Semi-honest model (passive adversary). Also called honest-but-curious model, this model was first introduced by Goldreich et al. [1987]. Parties in this model follow the computation protocol semi-faithfully, meaning that it can be unintentionally faulty in its computations but most importantly attempts to infer additional information from messages received during the protocol

execution. Since this model is based on the assumption that parties are honest, it may be unrealistic in some settings where a party is malicious.

Malicious model (active adversary). This model presents a more realistic case where a party deviates from the protocol. Among many other behaviors, a malicious adversary might inject false data or return false computation results in order to infer some sensitive information.

4.2.2 Applications

There are numerous applications of SPED. The most common are briefly listed as follows:

Electronic voting. Voting is a highly confidential activity, whether it involves a small group or a large nation. Privacy, integrity, and verifiability are essential requirements for a voting scheme to be accurate and secure. Privacy involves maintaining the anonymity of the vote, i.e., who voted for whom. Integrity ensures that all the votes are valid and there is no involvement of malicious voters. Verifiability is needed to cross-check the outcome of the votes. These requirements can be fulfilled by using zero-knowledge proofs and homomorphic cryptosystems [Benaloh 1994, Cramer et al. 1997, Damgard and Jurik 2001, Hirt and Sako 2000].

Function hiding. Function hiding involves encrypting the original function in such a way that anyone can do the computations themselves, keeping the function concealed from them. This reduces the huge transmission and communication cost between the creator and the user of the function. Sander and Tschudin [1998] have proposed utilizing the homomorphic encryption schemes to hide the function and let an unknown decrypt the result.

Digital watermarking. Digital watermarking involves covertly hiding information in a signal so as to identify its ownership. It has an important digital rights management application in verifying the authenticity or integrity of the original content (e.g., a user is allowed to run the original content, but not to copy it). Since the watermark is an inherent part of the signal, it should be kept secret or at least within a trusted environment [Cox et al. 1996]. In their 2001 article, Adelsbach and Sadeghi introduced a zero-knowledge watermark detection protocol. Kalker [2007] later proposed a more efficient variant of the same by using the Paillier cryptosystem.

Collaborative filtering. Personalized Internet usage has a huge impact on today's economy. Marketers use data obtained from digital footprints (i.e., social networking sites, search histories, etc.) to “personalize” the advertisements that the user sees on certain websites, suggesting products that correspond to the user's digital footprint. This “personalized” marketing, however, costs the users their privacy. Collaborative filtering in the encrypted domain can help allow users to keep control over their data while still being able to retrieve personal recommendations [Erkin et al. 2011]. Canny [2002] has proposed the same recommender system in encrypted domain.

Searching in encrypted databases. Traditionally, a keyword search is performed on the files kept in a database. However, in the current era of cloud computing, there is a continuous trend of delegating the multimedia processing tasks to third-party service providers. To maintain data confidentiality and integrity, there is an urgent need to keep multimedia data files in an encrypted format on the non-trusted servers, thereby carrying out secure processing of the content in encrypted domain [Song et al. 2000, Swaminathan et al. 2007, Lu et al. 2009a].

Processing encrypted multimedia files. In order to provide cloud computing storage and retrieval services, processing multimedia (in the form of image/video/audio content) in encrypted domain is inevitable from security and privacy perspectives. Several strategies have been proposed for secure face recognition in Avidan and Butman [2006] and Erkin et al. [2009], using homomorphic encryption. Upmanyu [2009, 2010], have proposed a framework for implementing video surveillance in encrypted domain using secret sharing techniques.

Matching biometric data. Privacy-preserving matching of biometrics is gaining importance, especially when a subject's identity is to be matched against a stored database of biometrics owned by an agency [Puech et al. 2012, Upmanyu 2010, Bianchi et al. 2010, Barni et al. 2010, Torres et al. 2014, Bringer et al. 2014, Pillai et al. 2011]. Encrypted domain processing is important for stopping an adversary from mis-utilizing a biometric system, which in turn can be used to access sensitive information by illegally impersonating the victim. Depending upon the level of security, encryption techniques can be applied to the subject's side, the data server's side, or both. In addition to thus securing the matching process, privacy-preserving techniques are also

being applied when securing the storage and transmission of biometric templates [Evans et al. 2015, Sutcu et al. 2007].

Secure processing of medical data. Processing of medical data (EEG, ECG, DNA, MRI, etc.) by remote health expert systems has been very valuable for telemedicine. However, there are some concerns about patient confidentiality due to the sensitivity of such records. SPED building blocks are being applied to make such systems secure and protect the privacy and confidentiality of patients. Examples are the work by Troncoso-Pastoriza et al. [2007], Szajda et al. [2006], and Kantarcioglu et al. [2008] on performing secure search over DNA data; work by Lazzeretti et al. [2012] on privacy-preserving for secure evaluation of the quality of electrocardiogram (ECG) signals by a remote health monitoring system in order to guarantee correct medical decisions; and work by Barni et al. [2011] on secure classification of ECG signals.

4.2.3 Benefits

The main advantage of SPED is that it allows for computing with concealed data. This provides a secure platform to the computing party, because the data remains obscured to the adversary (if any). This advantage allows for distributed computing and delegation of multimedia services. Hence, through this survey it is shown that there is a dire need for preserving the privacy of multimedia content (text/image/video/audio), especially in today's world when it is a trend to outsource the storage of data, and many high-end computations are done at an untrusted or a third-party server like a cloud data center.

The following are the major benefits of using SPED:

Data privacy and integrity. The processing of data in encrypted domain allows users to work on data in a concealed format. It allows for computing with this data in a hostile environment while protecting privacy and security. On the other hand, data integrity depends on the attacker model a security protocol addresses. Malicious models provide this requirement, whereas semi-honest models do not guarantee integrity, as they are based on assumptions that parties are honest and hence exclude integrity checks.

Data confidentiality. Encrypted domain processing of data saves a lot of communication and computation overhead. Since it involves direct processing of encrypted data, only the input and result have to be verified. The user

feels more confident about the correctness of results because he performs the processing tasks himself, thus achieving the goal of “security not at the cost of accuracy.” The performance of the encrypted domain implementation should be similar to that of the plaintext domain implementation.

Efficiency. Encryption and decryption are usually expensive. By directly processing data in encrypted domain, we can ensure that the computations will be securely performed, even in distributed third-party service providers, e.g., cloud data centers. Such untrusted servers are usually cheaper and can perform the high-end computing tasks efficiently.

Scalable solutions. Due to the latest trends in “big data” analytics, there are many huge data producing multimedia systems requiring distributed data storage and processing. By judiciously utilizing SPED, large amounts of multimedia data can be processed in a secure way on scalable platforms such as the cloud.

4.3 Image Processing in Encrypted Domain

We have seen that signal processing in encrypted domain was initially introduced by [Ahituv et al. \[1987\]](#), and after that, for more than a decade and a half, it was mainly concentrated in the textual domain. Recently, in the past half decade, movement has taken place and diverted the attention of researchers toward dealing with the specific problems encountered when processing other types of media (e.g., image, video, or audio) in encrypted domain, to maintain security and privacy. Image processing in encrypted domain has been applied in areas such as image search/retrieval, feature extraction, biometric recognition, and quality enhancement.

4.3.1 Image Search/Retrieval

There has been some work on typical (non-encrypted) multimedia retrieval in a secure way, evaluating the similarity of two files’ contents using the distance between their visual features, such as color histograms, shape descriptors, or salient points [[Datta et al. 2008](#)]. Other work by [Shashank et al. \[2008\]](#) has brought the attention of researchers to the problem of protecting the privacy of the query image when searching over a public database, where the query and database images are both kept in encrypted form. The authors formulated the query message and response message during multiple rounds of communications between the user and the server, such that the server is made oblivious to the actual search path and is

thus unaware of the query content. The technique used for secure image retrieval is analogous to PIR (private information retrieval) for secure document retrieval, and hence is called content-based information retrieval (CBIR) [Smeulders et al. 2000]. However, applying the cryptographic primitives to such CBIR systems is not straightforward; the distance between feature vectors after encryption is not preserved, and the efficiency and scalability are not easily achieved for multimedia retrieval. The comparison of the similarity metrics among high-dimensional vectors using cryptographic primitives is also complex. A survey of related challenges can be found in Lew et al. [2006].

In 2009, a paper by Lu et al. [2009a] followed the work by Swaminathan et al. [2007] for rank-ordered search over encrypted text documents, so that documents can be returned in the order of their relevance to the query term. Lu et al. thus introduced the multimedia community to the problem of enabling CBIR over encrypted multimedia (image/video/audio) databases. In this paper, the authors argued that secure text search techniques can be applied to “keyword based search of multimedia data.” Keyword search relies on having accurate text descriptions of the content already available, and its search scope is confined to the existing keyword set. On the other hand, a content-based search over an encrypted multimedia database, if it can be done, provides more flexibility; sample images, audio, or videos are presented as query, and documents with similar audio-visual content in the database are identified. This work was mainly restricted to processing encrypted images (scalable to large databases) and search indexes. The multimedia documents were first encrypted by the content owner and then stored onto the server (non-trusted). By jointly applying cryptographic techniques, such as OPE and randomized hash functions, with image processing and PIR techniques, secure indexing schemes were designed to provide both privacy protection and rank-ordered search capability. Two secure indexing schemes were proposed: the first scheme made use of inverted indexes of visual words and the second scheme exploited randomized hash functions. A thorough analysis of the retrieval results on an encrypted color image database and security analysis of the secure indexing schemes under different attack models show that data confidentiality can be preserved while retaining very good retrieval performance. A detailed description of the security analysis for a ciphertext-only attack and a known plaintext attack can be read in Lu et al. [2009a].

In another paper [Lu et al. 2009b], these authors presented complementary research to corroborate the aspects of encrypted multimedia retrieval for secure online services. The paper focused on image feature protection techniques (where data confidentiality is preserved both in the storage and retrieval processes) that

enabled similarity comparison among protected features. By utilizing both signal processing and cryptographic techniques (e.g., OPE, AES/RSA, etc.) along with image processing tools, three schemes have been investigated and compared by [Kim et al. \[2007\]](#), including bitplane randomization, random projection, and randomized unary encoding. This paper focused on the previously mentioned problem of image feature protection, which allowed the computation of similarity measures among encrypted features, so that secure CBIR [[Datta et al. 2008](#), [Shashank et al. 2008](#)] could be achieved. Thus, techniques for secure image retrieval to encrypt image features, while approximately preserving their distances and three feature protection schemes, are explored and compared in terms of security, retrieval performance, and computational complexity. A complete security analysis of [Lu et al. \[2009a\]](#) was then presented in [Lu et al. \[2010\]](#).

Another paper on secure multimedia (image and video) retrieval was presented recently in 2015 by [Chu and Chang \[2015\]](#), who propose a client-server privacy-preserving multimedia retrieval framework based on homomorphic encryption schemes (Paillier and DGK), garbled circuit, and bipartite graph. In their work, both the client's query (feature vectors of multimedia content-image or video) and the server's database are encrypted. The server then constructs a bipartite graph over the similarity between the encrypted query and database, and finds the minimum-cost bipartite graph matching by using the Hungarian algorithm in the encrypted domain. The advantage of this work over previous work [[Lu et al. 2009b](#)] on image retrieval is the use of a garbled circuit and packing scheme to reduce computation and communication cost, since the comparison protocol in ED was based on expensive computation on exponents and frequent interactions between client and server. However, there are minimal accuracy losses between encrypted domain and plaintext domain processing as a result of quantizing real values to integer.

4.3.2 Feature Extraction

Due to the fact that scale-invariant feature transform (SIFT) has been widely adopted in various fields (detecting and describing local features in images), in their 2011 article, [Hsu et al.](#) have sought to address the problem of secure SIFT feature extraction [[Hsu et al. 2009](#)] and representation in the encrypted domain. They propose a homomorphic encryption-based secure SIFT method for privacy-preserving feature extraction. To achieve secure SIFT, the Difference-of-Gaussian (DoG) transform is executed in the encrypted domain and a representation based

on the Paillier cryptosystem [Paillier 1999] is made. Moreover, to perform homomorphic comparison for SIFT feature detection, they investigated a quantization-like secure comparison strategy. Unlike the existing homomorphic cryptosystems providing additive and multiplicative homomorphism, the paper has remarkably presented the method to achieve local extrema extraction, descriptor calculation, and descriptor matching, all in the encrypted domain.

Later on in 2014, the secure SIFT comparison protocol proposed by Hsu et al. [2009, 2011], was analyzed by work in Schneider and Schneider [2014]. In that paper, the shortcoming of the comparison protocol is identified and potential remedy strategies are discussed. The authors argue that the weaknesses of the protocol are as follows. (i) Computation of the comparison protocol on the server is not feasible for large values of encrypted data (as a result of expansion due to increasing thresholds and large primes recommended by Hsu et al. [2009, 2011], when security parameters are chosen according to today's recommendations. Further proof can be found in Schneider and Schneider [2014]. (ii) The protocol, however, is computationally feasible when encrypted values are small, i.e., the modulus prime used for encryption is small (100 bits) as proposed in Hsu et al. [2009, 2011]. This results in reduced security, which leads to the second weakness. Based on this reduced security, the paper provides scenarios for cryptanalysis on the protocol. In order to achieve a balance between computational feasibility and security, the paper suggests alternative comparison protocols utilizing (1) interactive protocols with additively homomorphic encryption where interaction between user and server is minimized by outsourcing the computation to two non-colluding servers, or (2) non-interactive comparison using fully homomorphic encryption. In an attempt to address the unrealistic computational problem of comparison protocol in Hsu et al. [2011], Qin et al. [2014] have designed a cloud-based architecture for the secure computation of SIFT. In contrast to utilizing computationally expensive homomorphic encryption (Paillier) as in Hsu et al. [2011], their scheme leverages the efficiency of splitting-based encryption, order preserving encryption (OPE), random permutation, and dummy-point perturbation as their cryptographic primitives. Furthermore, the authors proceed to highlight the feasibility of their framework in a real-world setting by deploying it on the Microsoft Azure cloud environment.

Work by Kiya and Fujiyoshi [2012] presents a framework for signal and image processing in encrypted domain. Their work applies directly to the contemporary cloud computing framework and facilitates the transmission and processing of multimedia (mainly image) content in encrypted form. Their motivation lies in

the utilization of basic additive and multiplicative homomorphic properties for image processing in encrypted domain. They support their ideas with the help of two tangible examples:

- Demonstrating the DCT sign correlation for images, about which contains important information its corresponding signal and is useful for identification and estimation of displacement amount, rotation angle, and scaling factor, similar to the phase correlation. In order to perform DCT in encrypted domain, the authors first apply DCT to the original signal to obtain DCT coefficients and then the coefficients are separated to corresponding magnitude and sign. By using cryptographic public encryption, a sequence is generated by a stream cipher with a key, and signs are encrypted, thereby maintaining a correlation in encrypted domain.
- Identifying JPEG 2000 images in encrypted domain. The authors first encode images/video sequences by JPEG 2000; then they are encrypted and stored in a database. A query image is also encoded by JPEG 2000, and then compared to compressed-and-encrypted images in the database (a detailed description can be read in [Kiya and Fujiyoshi \[2012\]](#)).

4.3.3 Biometric Recognition

In work by [Yogachandran et al. \[2012\]](#), the authors perform encrypted domain facial expression recognition, based on local Fisher discriminant analysis over the cloud. A framework based on an asymmetric cryptosystem is suggested, in which the client distributes its public key to the server (at a cloud data center) and keeps its private key a secret. The server is able to perform encryptions under this public key and processes the data in encrypted domain. However, only the client is able to decrypt any encrypted messages using its corresponding private key. The processing of data in encrypted domain utilizes the additive homomorphic properties of the Paillier cryptosystem to perform the required linear operations. In recognizing up to 95.24% of the facial expressions computed, this system outperforms the chosen database in encrypted domain.

4.3.4 Quality Enhancement

In [Bogdanov \[2007\]](#) and [Islam et al. \[2009\]](#), authors study the homomorphic properties of secret sharing to perform mathematical operations on the share data itself. Recently, there have been a few related studies involving direct processing of the

encrypted share images using secret sharing [SaghaianNejadEsfahani et al. 2012], performing image denoising in wavelet domain. In their 2012 article, Mohanty et al. first suggested utilizing the additive and multiplicative homomorphism to perform secure rendering on secret-shared medical images over third-party cloud data centers. They later extended this work to show scaling and cropping operations over medical images [Mohanty et al. 2013]. This was clearly a breakthrough in using real number analysis for addition and multiplication over images in encrypted domain. Furthermore, both of their papers addressed a security requirement that ensured data integrity and protection against tampering with medical data over the cloud. Other noteworthy work by Lathey et al. [2013] first presented the possibility of obtaining equivalent results in encrypted and plaintext domain(s), involving division operations (especially involving non-terminating quotients) for low-pass filtering of images over the cloud. Later, these authors extended their research to propose schemes for secure image enhancement (anti-aliasing, edge sharpening, contrast enhancement, and dehazing) over the cloud [Lathey and Atrey 2015]. Their secure implementation was based on performing unsharp masking and histogram equalization over encrypted image shares over the cloud. Security of their work was proven from the perspective of preservation of information theoretic security from (T, N) SSS.

There has not been much work with the application of fully homomorphic encryption (FHE) techniques for privacy-preserving of multimedia content. However, recent work proposed by Shortell and Shokoufandeh [2015] would provide an implementation of a brightness/contrast filter for images in encrypted domain with FHE. The architecture of the proposed work is based on a two-party scenario where the security and privacy of a client's signal is protected while being processed on a third-party server. Authors discuss the limitations of their work as follows: (i) numerical errors are introduced into the secure filters operation when compared to PD processing as a result of scaling real-valued filter coefficients to integer values, and (ii) FHE is computationally expensive; for this reason authors state that a GPU can be used to improve performance.

A summary of all the papers described above is provided in Table 4.2. It is clear that by adopting various tools/techniques utilized for processing encrypted text content, powerful computing can also be done on image data in encrypted domain. For example, searching for images over encrypted databases using homomorphism (Paillier cryptosystem), randomized hash function, homomorphic secret sharing, etc., can be treated as analogous to secure text data processing in the encrypted domain.

Table 4.2 A list of the representative literature in the encrypted domain image processing category

Papers	Processing task	Technique used	Data overhead	Computation overhead	Attacker model
Lu et al. [2009a]	Keyword-based search for encrypted images	OPE using randomized hash functions	High	Low	Semi-honest
Lu et al. [2009b]	Secure image retrieval using bitplane randomization, projection, and unary encoding	OPE, RSA/AES	High	High	Semi-honest
Hsu et al. [2011]	Homomorphic SIFT feature extraction, local extrema and descriptor calculation and matching	Homomorphism, Paillier cryptosystem	Medium-High	High	Malicious
Kiya and Fujyoshi [2012]	DCT sign correlation and identifying JPEG 2000 images	Homomorphism, stream cipher sequences	High	Low	Not mentioned
SaghaiNejadEsfahani et al. [2012]	Image denoising in wavelet domain	Secret sharing	High	Low	Semi-honest
Yogachandran et al. [2012]	Cloud-based facial expression recognition based on local Fisher discriminant analysis	PRK, Paillier cryptosystem	High	High	Not mentioned
Mohanty et al. [2012]	Rendering of medical images	Ramp secret sharing	High	Low	Malicious
Mohanty et al. [2013]	Scaling and cropping of medical images	Ramp secret sharing	High	Low	Malicious
Lathey et al. [2013]	Division operations for low-pass filtering over the cloud	(T,N) - SSS	High	Low	Semi-honest
Lathey and Atrey [2015]	Secure image enhancement over the cloud	(T,N) - SSS	High	Low	Semi-honest
Qin et al. [2014]	Secure SIFT feature extraction over cloud	Secret sharing and OPE	High	Low	Semi-honest
Shortell and Shokoufandeh [2015]	Implementation of a brightness/contrast filter in ED	FHE	High	High	Not mentioned
Chu and Chang [2015]	Secure image retrieval using bipartite graph matching	Homomorphic encryption schemes (Paillier and DGK)	Medium-High	Low	Semi-honest

4.4 Video Processing in Encrypted Domain

Recent tremendous growth in video data in various scenarios, including CCTV surveillance, social media and personal video albums, has posed challenges for its secure storage and processing. There has been a continuous trade-off between privacy and security, and many research efforts have been made to explore the ways in which privacy comes to the fore in day-to-day applications where video is captured. For example, in public video surveillance systems [Slobogin 2002, Bennett 2008, Norris et al. 1998, Walby 2005, Dubbeld 2002, Hubbard et al. 2004], it is well established that these systems have been very useful for public safety, but the widespread usage of surveillance cameras at offices, hospitals, parks, streets, shopping malls, parking lots, residential complexes, schools, banks and other business/public/commercial establishments raises privacy concerns in watching the captured people's private moments, location, and companion(s), leading to further analyzing their time-stamps and associated day-to-day activities.

Yet "necessity is the mother of invention." These privacy concerns pointed toward developing schemes and architectures for secure and privacy-aware video-based systems. Thus, having created a niche in developing tools for processing encrypted text and image data, researchers have started to harness the unexplored opportunities available in processing encrypted video data (particularly, in surveillance applications). Although it is in a nascent stage, this area of research is one of the most sought-after among the researchers in the multimedia community. Video processing in encrypted domain has been applied in various ways, such as by making video data unrecognizable, exploring the new secure domains for video encoding, and implementing secure video processing in encrypted domain.

4.4.1 Making Video Data Unrecognizable

There have been several attempts to protect video data by making video content unrecognizable. A brief history summarizing privacy issues in surveillance videos is presented in Upmanyu et al. [2009]. The authors point out that SMC has been used initially as a privacy-preserving cryptographic protocol [Shashank et al. 2008]. In fact, Avidan and Butman [2006] have proposed the idea of blind vision, allowing a classifier to run on someone's data without revealing the algorithm or gaining knowledge of the data. Masking only the sensitive information in the videos (like faces) has also been studied in Chattopadhyay and Boulton [2007], Spindler et al. [2008], Boulton [2005], Martínez-Ponte et al. [2005], and Dufaux and Ebrahimi [2004]. The efficiency of such schemes was related to the combination of classical

cryptographic tools/techniques with modern video processing methods. Only an authorized user possessing the secret key could have access to the real data. Furthermore, selective scrambling to conceal the ROI in MPEG 4 surveillance videos, using a new transform domain–codestream scrambling method has been introduced in [Dufaux and Ebrahimi \[2008\]](#). Prior to their work, [Zeng and Lei \[1999, 2003\]](#) and [Wen et al. \[2001, 2002\]](#) proposed scrambling techniques in the frequency domain by employing selective encryption of compressed videos' bitstreams and shuffling. Face swapping [[Bitouk et al. 2008](#)] (for similar face poses using their built-in face library for replacing similar skin tones, lighting conditions, and viewpoints) and face [[Newton et al. 2005](#)] (using their own privacy protecting *k-same* algorithm) or person [[Luo et al. 2010](#)] (using homomorphic encryption) deidentification has also been proposed as a solution to preserve privacy. Video data transformation by a combination of quantization and blurring has been proposed by [Saini et al. \[2013\]](#), in order to get the appropriate trade-off between privacy and utility. Some other layered encoding-decoding architectures have been put forward to hide the privacy-intrusive details while preserving the information necessary for the system to be useful [Senior et al. \[2003\]](#), [Iqbal et al. \[2006\]](#). The ROI-based control scheme for encoding traffic surveillance videos has also been proposed [[Ching and Su 2009](#)].

Hence a summary of the literature concerning protecting the privacy of sensitive issues in videos (particularly, surveillance data) can be seen in Table 4.3.

4.4.2 Exploring Secure Domains for Video Encoding

All the above approaches (except [Saini et al. \[2013\]](#)) rely on the successful detection of the ROIs and do not provide any guarantee of achieving perfect security and privacy. Moreover, the original video is lost in most of the cases after applying privacy-preserving measures. However, an important paper by [Mao and Wu \[2006\]](#) discusses the importance and feasibility of applying a joint signal processing and cryptographic approach to multimedia encryption, in order to address the access-control issues unique to multimedia applications. The goal of the authors was to design encryption tools for multimedia that could encrypt once and securely process in many ways using existing multimedia signal processing techniques. In particular, they investigated the possible domains in which encryptions can be applied, including the sample domain, the quantized transform domain, the intermediate bitplanes, and the bitstream domain. The resulting system takes into consideration the structure and syntax of multimedia sources and protected the

Table 4.3 A list of the representative literature for protecting privacy of selective regions in videos

Papers	Processing task	Technique used
Shashank et al. [2008]	Retrieval of images/videos	PIR, hashing/indexing for tree-based searches
Avidan and Butman [2006]	Blind vision: video data classifier	SMC
Chattopadhyay and Boulton [2007]	Video surveillance	Programmable cameras
Spindler et al. [2008]	Detection of ROI	Programmable cameras
Zeng and Lei [1999] and Zeng and Lei [2003]	Selective scrambling	Selective bit scrambling, block shuffling, and block rotation
Wen et al. [2001] and Wen et al. [2002]	Selective scrambling	Selective encryption and spatial/frequency shuffling
Dufaux and Ebrahimi [2008]	Selective scrambling	Transform domain encryption
Bitouk et al. [2008]	Face swapping	Face replacement using built-in face library
Newton et al. [2005]	Face deidentification	Proposed <i>k-same</i> algorithm
Saini et al. [2013]	Surveillance video	Quantization and blurring
Senior et al. [2003] and Iqbal et al. [2006]	Layered encoding-decoding architectures	Key-based cryptographic protocols

content confidentiality during delegate processing. The mathematical details of all these operations can be found in Mao and Wu [2006].

An interesting contribution of this work led is to introduce a notion of multimedia-oriented security to evaluate it against approximation recovery. They use the security of visual data as an example and have proposed a detailed version of their visual security scores (extendable to auditory and other types of multimedia). They also presented a framework for a video encryption system. Using this system, several example configurations were presented and the encrypted videos were compared in terms of the security against brute-force and approximation attacks, the friendliness to delegate processing, and the compression overhead. In addition to this, the experimental results show that by strategically integrating selective value encryption and intra-bitplane shuffling, as well as spatial permutation, the resulting encryption system could achieve a good trade-off among security, friendliness to delegate processing, and bit-rate overhead.

4.4.3 Implementing Secure Video Processing in Encrypted Domain

Following the trend, the work by Upmanyu [2009, 2010] has proven to be important research related to “efficient privacy preserving protocols for visual computation.”

These authors have purely focused on the development of highly secure communication and computationally efficient algorithms for problems with “immediate impact” in the domain of computer vision and related areas. They studied and proposed secure solutions for applications such as blind authentication, i.e., blindly authenticating a remote-user using his biometric, object tracking, and face detection over the cloud. They presented a secure framework for carrying out visual surveillance on random-looking video streams at remote servers. Certain desirable properties of visual data such as fixed range and insensitivity to data scale were explored to achieve distributed, efficient, and secure computation of surveillance algorithms in the above framework. However, this approach had two shortcomings: (i) the method was useful for carrying out integer addition and subtraction operations only, and (ii) the authors themselves stated that their system was not able to perform the division operations, as it required choosing a prime number in such a way that the size of the modulo domain would be increased more than required. Subsequently, an alternative was proposed by using an additional computation server where the merge function was applied to respective residues obtained from other independent servers and the division/comparison was performed in the real domain. Hence, it led to another requirement to secure the intermediate information against attacks. Also, there was an additional communication overhead in sending the residues to the additional server and receiving the processed data back for this purpose. Recently, [Chu et al. \[2013\]](#) have proposed a parallel aspect of real-time privacy-preserving moving object detection in the cloud by utilizing homomorphic properties of the Paillier cryptosystem.

Furthermore, [Erkin et al. \[2009\]](#) propose encrypting face images using homomorphic encryption and letting the eigenface recognition algorithm work on encrypted data without revealing private information to the holder of the face database. [Sadeghi et al. \[2010\]](#) have further improved the efficiency of the proposed approach by replacing the matching mechanism in [Erkin et al. \[2009\]](#) with a fine-tuned garbled circuit. [Sohn et al. \[2010\]](#) proposed watch list screening for video surveillance systems that discriminates groups of identities of interest without revealing face images. [Osadchy et al. \[2010\]](#) have proposed a new face identification system designed for use in secure computation based on homomorphic encryption and oblivious transfer protocols.

Similar to the deduplication technique on encrypted text data in [Fan et al. \[2015\]](#), [Zheng et al. \[2015\]](#) have also sought to optimize transmission and storage of video over the cloud in a privacy-preserving manner. Their proposed scheme performs deduplication on encrypted SVC (scalable video coded) videos. The goal of this work is to greatly reduce the network bandwidth and eliminate the storage redundancy of

SVC video files in cloud services to ensure fast retrieval and efficient dissemination to heterogeneous networks and different devices. Their architecture involves a user (client), a third-party agency, and the CDC. The user (client) engages in an RSA-OPRF protocol with the agency in a blinded manner to derive message tags of the various layers of the SVC video before uploading to the cloud for duplication checks in a secure manner. Authors utilize an RSA-OPRF scheme built on RSA blind signatures to meet security requirements of their framework. This achieves security and confidentiality for both semi-honest and malicious adversarial cases over the cloud in the bounded leakage setting proposed by [Xu et al. \[2013\]](#), to defend against off-line brute-force attacks over predictable videos.

A summary of all the literature related to encrypted domain video processing is provided in [Table 4.4](#). It is clear that most of the previous work has been done for video processing in secure domain, i.e., by selective encryption or performing some irreversible operations on the video data, due to which the video content becomes unrecognizable. The drawback of using such techniques is that original video content is also lost. Hence, in the past 4–5 years, researchers have started to process video data directly in encrypted form. So far, only a few authors demonstrate a secure privacy-aware system for carrying out video processing tasks in an efficient manner.

4.5 Audio Processing in Encrypted Domain

Unlike image and video processing in the encrypted domain, the processing of encrypted audio signals has been least explored. Work in this area has focused on two broad application domains: (i) speech/speaker recognition—here feature vectors are extracted from an encrypted speech signal and processed or trained for the purpose of identification and recognition—and (ii) audio editing/quality enhancement—here encrypted audio signals are processed in time or other domains (FFT, wavelet etc.) to bandpass or stop certain frequencies that allow for the manipulation of properties such as pitch, amplitude, timbre, phase shifts etc. Adjusting these properties makes it possible for audio to be edited or enhanced in quality.

4.5.1 Speech/Speaker Recognition

Speech/speaker recognition in encrypted domain involves the processing of compressed voice over Internet protocol (VoIP) traffic over communication channels. The work in this area began with [Rose and Paul \[1990\]](#) and [Wilpon et al. \[1990\]](#), who invented keyword recognizers (KWR) based on hidden Markov models (HMM)

Table 4.4 A list of the representative literature in encrypted domain video processing

Papers	Processing task	Technique used	Data overhead	Computation overhead	Attacker model
Mao and Wu [2006]	Signal processing and cryptographic approach to multimedia (mainly video) encryption	Proposed own encryption tools using index mapping and intra-bitplane shuffling	Medium-high	High	Not mentioned
Upmanyu [2010] and Upmanyu et al. [2009]	Efficient privacy preserving protocols for visual computation in biometric authentication, change detection, optical flow, and face detection	Secret sharing (CRT based), SMC	High	Low	Semi-honest
Sohn et al. [2010]	Watch list screening for video surveillance systems	Homomorphic encryption, PRKC	High	Low	Semi-honest
Erkin et al. [2009]	Secure face recognition	Homomorphism, Paillier cryptosystem	High	Low	Semi-honest
Sadeghi et al. [2010]	Encryption of facial images	OT, Homomorphic encryption using [Damgard and Jurik 2001]	High	Low	Semi-honest
Osadchy et al. [2010]	Face identification system	Homomorphic encryption, OT	High	Low	Semi-honest
Chu et al. [2013]	Real-time privacy-preserving moving object detection in the cloud	Homomorphism, Paillier cryptosystem	High	Low	Not mentioned
Zheng et al. [2015]	Deduplication over encrypted SVC (Scalable Video coded) videos	RSA with oblivious pseudo-random function	Low	High	Semi-honest and malicious

for keyword search on speech over a communication channel. Their techniques recognized a predefined set of keywords within a long-distance telephone network. Since then, there has been continuous work toward the identification of people with malicious intent talking over VoIP systems [Moreno et al. 2001, Wright et al. 2008].

Certain properties of encrypted and non-silence-suppressed VoIP traffic such as packet size, packet timing, and bit rate, show some correlation with speech activity. Due to this fact, some literature suggests that encrypted or compressed VoIP traffic is not fully secure [Saponas et al. 2007]. These properties can be exploited to develop algorithms to distinguish encrypted or compressed VoIP traffic containing speech activity from other activity such as web traffic or online game traffic. Almost all work in the area of speech activity identification in encrypted or compressed VoIP is based on this idea. For instance, Chang et al. [2008] leveraged the packet size of VoIP traffic to detect the presence or absence of speech activity in it. Aggarwal et al. [2005] have presented a compressed speaker recognition (CSR) system to identify a speaker directly from compressed VoIP packets, contrary to the traditional system where the compressed voice packets had to be decompressed first. CSR utilized a micro-clustering algorithm proposed by Aggarwal et al. [2003] to obtain accuracy three times higher than the frequently used Gaussian mixture model (GMM). Using packet timing for VoIP traffic, Wang et al. [2005] have presented a watermark technique for tracking anonymous peer-to-peer VoIP calls on the Internet. They further presented an in-depth analysis of how low-latency anonymizing networks were susceptible to timing attacks.

An attempt to identify a caller/callee pair of streams was made by Verscheure et al. [2006]. The authors exploited the aperiodic inter-departure time of VoIP packets to trivialize each VoIP stream into a binary time-series, followed by a progressive clustering procedure. This led to the design of a metric to gauge the correlation between two VoIP binary streams, thereby showing how to reveal a pair of anonymous and encrypted conversing parties. Wright et al. [2007] explicitly used the length of encrypted VoIP packets to identify the language of conversation. They were able to successfully identify 14 of 21 languages, with an accuracy of more than 90%. Following this, the same authors showed in 2008 that it is possible to identify the phrases spoken within a call with the help of the lengths of encrypted VoIP packets, when variable bit rate codecs are used for encrypting the audio. They utilized the properties of reducing speech to phonemes, proposed the method of encrypted VoIP speech recognition as a sub-string matching problem, and reconstructed sentences using HMMs, with an accuracy of 50% (for standard speech corpus) to 90% (for some phrases). All the researchers discussed above have exploited the

Table 4.5 A list of the representative literature in processing un-encrypted/compressed VoIP signals

Papers	Processing task	Technique used
Rose and Paul [1990]	Keyword recognizer for speech	HMM
Wilpon et al. [1990]	Vocabulary keywords for speech	HMM
Wang et al. [2005]	Tracking anonymous peer-to-peer VoIP calls	Packet timing as a watermarking
Aggarwal et al. [2005], Aggarwal et al. [2003]	Speaker identification directly from the compressed voice packets	GMM, micro-clustering algorithm
Chang et al. [2008]	Network-level VAD	Adaptive thresholding
Verscheure et al. [2006]	Identifying caller/callee pair over VoIP	Packet timing and clustering algorithm
Wright et al. [2007], Wright et al. [2008]	Identifying language and phrases in encrypted speech over VoIP	Sub-string matching algorithm and HMM

vulnerabilities inherent to VoIP systems to either detect speech activity or the language spoken and did not process audio/speech in encrypted domain. An overview of various threats and vulnerabilities in the area of securing VoIP packets can be found in Keromytis [2009]. Table 4.5 summarizes the literature related to processing un-encrypted/compressed speech signals.

In contrast with these techniques, Khan et al. [2010] is among the few to propose the actual implementation of direct processing of encrypted VoIP packets for speaker identification and verification in encrypted domain. Encoding of VoIP traffic to narrow band prior to encryption is a common practice to save transmission bandwidth. Techniques such as variable bit rate (VBR) encoding used in real-life scenarios result in variable-length VoIP packets. There is a relationship between this length and a speaker's identity, which remains unchanged even after encryption with secure real time transport protocol (SRTP) based on the Advanced Encryption Standard (AES). The basic idea behind this work stems from observing the relationship between the speaker's identity and the length of the packet carrying their VoIP speech contents. Hence by utilizing discrete HMM and GMM to create models for each speaker based on the sequence of the packet-length extracted from encrypted VoIP conversations, Khan et al. [2010] were able to achieve the goal of speaker identification and verification from encrypted VoIP packets. Subsequently, by using VAD instead of VBR, Backes et al. [2010] provided the same kind of study with comparable results for speaker identification in encrypted domain for VoIP packets. From a security perspective, neither paper formulated any security model and requirements for their work; however, they did demonstrate that VoIP commu-

nications are not secure. And by exploiting these vulnerabilities, computation on encrypted VoIP traffic can be achieved.

[Pathak et al. \[2013\]](#) and [Shashanka and Smaragdis \[2006\]](#) have also proposed a framework for speaker verification/identification and sound recognition/classification, respectively, using GMM and likelihood ratio test in encrypted domain. Both methods are based on SMC and homomorphic encryption (Paillier and Boneh-Goh-Nissim (BGN) cryptosystem) that enables computation and classification to be performed in a secure way. [Pathak et al. \[2013\]](#) have proposed a client-server architecture where the client has a speech sample and the server stores the encrypted model (GMM parameters of the speech) after the training process. For a secure speaker recognition/verification to be performed, the client sends the encrypted feature vectors (mel-frequency cepstral coefficients) of the speech sample to the server, which then computes the inner products between the encrypted feature vectors and the encrypted GMM models, using the homomorphic properties of the Paillier and BGN cryptosystem to obtain a score. The score is then compared with a threshold to determine whether there is a match or not. The authors utilized secure comparison protocols (secure maximum index protocol and Yao millionaire protocol) for the matching process. [Shashanka and Smaragdis \[2006\]](#) have applied the same approach for secure classification of sound. Security requirements of both [Pathak et al. \[2013\]](#) and [Shashanka and Smaragdis \[2006\]](#) are formulated to address both semi-honest and malicious adversarial models.

4.5.2 Audio Editing Quality Enhancement

All work on audio/speech processing in encrypted domain had focused on classification problems for recognition and verification until 2015, when [Yakubu et al. \[2015\]](#) proposed for the first time the secure addition of reverberation effect to an audio signal stored on the cloud for the purposes of editing and reproduction. Their work was based on convolution of the impulse response of an acoustic space with the encrypted audio signals over the cloud. The playback of the edited signal after decryption sounded as though it had been recorded in that acoustic space.

A summary of the literature involving processing encrypted audio signals is given in Table 4.6. This area of multimedia processing seems to be least explored. Also, it is specifically mentioned by [Khan et al. \[2010\]](#) that the current state of the art in speaker recognition techniques (whether in encrypted or un-encrypted domain) have not matured enough to be presented in a court as the sole source of evidence against a suspect; these techniques are just complementary to forensic investigations.

Table 4.6 A list of the representative literature in encrypted domain audio processing

Papers	Processing task	Technique used	Data overhead	Computation overhead	Attacker model
Khan et al. [2010]	Speaker identification and verification in encrypted domain using VBR	AES	High	High	Not mentioned
Backes et al. [2010]	Speaker identification and verification in encrypted domain using VAD	AES	High	High	Not mentioned
Pathak et al. [2013]	Speaker identification and verification in encrypted domain	SMC and homomorphic encryption (Paillier and BGN cryptosystem)	Medium-high	High	Semi-honest and malicious
Shashanka and Smaragdīs [2006]	Sound recognition and classification in encrypted domain	SMC and homomorphic encryption (Paillier and BGN cryptosystem)	Medium-high	High	Semi-honest and malicious
Yakubu et al. [2015]	Addition of reverberation effect to an audio secret over the cloud	(T,N) - SSS	High	Low	Semi-honest

4.6 Further Discussion

4.6.1 Future Research and Challenges

We have seen a number of encrypted domain multimedia analysis tasks in the literature; however, there are several open research problems that need investigation in the future. We have identified some of them as follows:

Data overhead. For security reasons, homomorphic encryption schemes are probabilistic in nature, which means that for a particular encryption key, each plaintext may result in several different cipher text values. They may also require long security parameters (e.g., key size) to protect against brute force attacks, which also increases the ciphertext space. This increase in the size of the ciphertext space leads to huge data expansion when transmitting the encrypted data to the third-party service providers. This also results in an increased computational complexity. The resulting systems have to be specified as computationally secure or unconditional/information theoretically secure.

Computational complexity. Suitable adaptation of the cryptographic tools to perform multimedia specific processing tasks needs to be examined carefully. A slightly expensive scheme can be utilized for most of the offline multimedia processing, but for real-time analysis in encrypted domain, a highly computationally efficient scheme is desired. Thus, it is quite a daunting task to choose the one suitable for one's needs.

Algebraic homomorphic property. Until 2009 there were only a few authors [[Gentry 2009](#), [Dijk et al. 2010](#)] underscoring the usage of a secure encryption scheme with an algebraic homomorphic property. This limited the possible applications of SPED, because they could only be applied to problems which require either additive or multiplicative homomorphism. However, one of the recent noteworthy papers by Gentry presents a more formal fully homomorphic encryption scheme, Brakerski-Gentry-Vaikuntanathan (BGV) [[Brakerski et al. 2012](#)]. This scheme is based on lattices and deals with integer vectors and polynomials as well. However, there are still many applications to explore, utilizing this encryption scheme. Also, the authors admit that although the scheme has strong information theoretic security, a detailed analysis still needs to be done.

Complex mathematical operations. Problems related to representing real numbers/negative numbers in secure domain, overflow and underflow, integer division and thresholding, and defining equivalent operations [Upmanyu 2010] analogous to the operations performed in plaintext domain (e.g., calculation of local maxima or minima of numbers in encrypted domain) remain challenges facing researchers. Although Brakerski et al. [2012] suggest the possibility of many of the image processing operations in encrypted domain, a thorough analysis of the scheme is needed. Also, there is research [Franz and Katzenbeisser 2011, Franz et al. 2010, Franz et al. 2010] that proposes implementations of real numbers (floating point arithmetic IEEE 754) in encrypted domain; however, further optimizations are still required.

Security vs. accuracy. Making the system perform equivalent operations in both encrypted and plaintext domains always poses the dilemma of a trade-off between the security of data and the accuracy of the data processing task. Having a balance between the two contradictory goals is challenging.

Under-explored area of encrypted domain audio processing. As detailed earlier, audio processing in encrypted domain has been limited to mostly speaker identification or keyword recognition with lower accuracies of the operations in encrypted domain when compared to their corresponding ones in the plaintext domain. There remains a plethora of opportunities for the new researchers in SPED to examine and enhance the efficiency/performance of such systems and explore the feasibility of other higher-level audio semantic analysis tasks in encrypted domain.

Processing of other multimedia in encrypted domain. There might be some possibilities in adapting the concept of SPED to other multimedia like RFIDs, graphic models and designs, motion sensors, etc. It would be quite interesting to find how the fusion of two or more multimedia components will work in encrypted domain, effectively achieving the highest goals of privacy and security.

Performing encrypted domain multimodal analysis tasks. So far, most of the encrypted domain analysis tasks have been performed on a single medium. It remains to be seen how the current methods can be used to perform encrypted domain analysis tasks involving multiple media.

Other emerging applications. It would really be a difficult task to suggest or spontaneously analyze the plausibility of a given application in encrypted domain. There are some emerging applications, depending on the need to

protect the privacy of user-related information, for processing encrypted signals [Puech et al. 2012]. For example, in privacy-aware smart electricity grids [Castelluccia et al. 2009, Tudor et al. 2015], load balancing and price negotiations can be performed by the energy distributor in encrypted domain. This nullifies attempts to infer users' behavior from the observed energy demand, using smart meters for load balancing in the energy network and real-time energy price negotiations.

4.6.2 A Ready Reckoner

This subsection provides a guide to preemptive measures for naive and future researchers in this emerging field:

Break the complex mathematical operations into simpler ones. Fit the complex multimedia processing operations into building blocks, especially with regard to additive or multiplicative homomorphism. For instance, try breaking the differential operations into step-wise subtraction or making real number multiplication/division possible by doing some pre-processing of the original multimedia data.

Decide between computational/conditional or information theoretic/unconditional security. In order to process data for a specific task, at times some of the patterns/coherence sequences have to be maintained. Once the data is encrypted using the information theoretic cryptosystem, all the coherence is lost. And, if a computationally secure cryptosystem is used, the homomorphic property may allow for tampering with ciphertexts, so that the cryptosystem could be broken in limited attempts and the result would be influenced. Thus, a lot of care needs to be taken while designing a tamper-proof scheme.

Trade-off between privacy/security and accuracy. Accept limited general information leakage only. It is important to gain the most efficiency possible in performing a task in encrypted domain, but not at the cost of accuracy. Adding extra randomness to prevent information leakage and holding privacy/security measures in place is quite acceptable.

Computational efficiency and data integrity. Try to reduce computational complexity by performing some of the tasks as preprocessing/postprocessing operations. However, ensuring data integrity (i.e., checkability and verification of computation results in encrypted domain) comes with additional

computational cost and/or multiple rounds of communication; e.g., implementing security models to address malicious adversaries is much more expensive and complex than honest-but-curious adversaries. Hence the requirements for security should evaluate the need for integrity checks alongside the additional complexity and its feasibility.

Transmission efficiency. Since the process of encryption leads to an increased ciphertext space, there must be some trade-off between data expansion and underlying security schemes. The goal should be to achieve minimal data overhead with acceptable security needs.

4.7 Conclusion

In this survey, we have attempted to provide readers with an overview of traditional and contemporary research areas of multimedia (text, image, video, and audio) processing in encrypted domain. It is one of the most sought-after areas of research for secure processing of multimedia data. Researchers have been successful in achieving various useful tasks in encrypted domain: data retrieval and processing, computer vision, semantic data analysis, mining, etc. Homomorphism [Legendijk et al. 2013] has always remained the “heart and soul” of multimedia signal processing tasks that can be performed with near-zero loss in accuracy in both plaintext domain and encrypted domain. Most of the homomorphic operations are defined for basic arithmetic (additive and multiplicative operations only). There is a lack of cryptographic schemes to perform high-level mathematical operations (maxima, minima, etc.) in encrypted domain. However, some serious efforts have been made in the past 4–5 years to invent fully homomorphic encryption schemes (e.g., BGV). The in-depth security analysis of such schemes is yet to be performed and proved to be applicable to higher-level mathematical operations.

Hence, there is tremendous scope for research in the area of developing fully homomorphic cryptosystems that can be directly applied for complicated mathematical operations to process multimedia data in encrypted domain. We are hopeful that the literature discussed as the latest developments in the area will contribute toward achieving many more crucial research milestones in this field.



Efficient Similarity Search

Hervé Jegou (Facebook)

This chapter addresses one of the fundamental problems involved in multimedia systems, namely efficient similarity search for large collections of multimedia content. This problem has received a lot of attention from various research communities. In particular, it is a historical line of research in computational geometry and databases. The computer vision and multimedia communities have adopted pragmatic approaches guided by practical requirements: the large sets of features required to describe image collections make visual search a highly demanding task. As a result, early works [[Flickner et al. 1995](#), [Fagin 1998](#), [Beis and Lowe 1997](#)] in image indexing have foreseen the interest in approximate algorithms, especially after the dissemination of methods based on local description in the 90s, as any improvement obtained on this indexing part improves the whole visual search system.

Among the existing approximate nearest neighbors (ANN) strategies, the popular framework of Locality-Sensitive Hashing (LSH) [[Indyk and Motwani 1998](#), [Gionis et al. 1999](#)] provides theoretical guarantees on the search quality with limited assumptions on the underlying data distribution. It was first proposed [[Indyk and Motwani 1998](#)] for the Hamming and ℓ_1 spaces, and was later extended to the Euclidean/cosine cases [[Charikar 2002](#), [Datar et al. 2004](#)] or the earth mover's distance [[Charikar 2002](#), [Andoni and Indyk 2006](#)]. LSH has been successfully used for local descriptors [[Ke et al. 2004](#)], 3D object indexing [[Matei et al. 2006](#), [Shakhnarovich et al. 2006](#)], and other fields such as audio retrieval [[Casey and Slaney 2007](#), [Ryynanen and Klapuri 2008](#)]. It has also received some attention in a context of private information retrieval [[Pathak and Raj 2012](#), [Aghasaryan et al. 2013](#), [Furon et al. 2013](#)].

A few years ago, approaches inspired by compression and more specifically quantization-based approaches [Jégou et al. 2011] were shown to be a viable alternative to hashing methods, and shown successful for efficiently searching in a billion-sized dataset.

This chapter discusses these different trends. It is organized as follows. Section 5.1 gives some background references and concepts, including evaluation issues. Most of the methods and variants are exposed within the LSH framework. It is worth mentioning that LSH is more of a concept than a particular algorithm. The search algorithms associated with LSH follow two distinct search mechanisms, the probe-cell model and sketches, which are discussed in Sections 5.2 and 5.3, respectively. Section 5.4 describes methods inspired by compression algorithms, while Section 5.5 discusses hybrid approaches combining the non-exhaustiveness of the cell-probe model with the advantages of sketches or compression-based algorithms. Other metrics than Euclidean and cosine are briefly discussed in Section 5.6.

5.1 Background

The objective of similarity search is to find “neighbors” of a given query vector $x \in \mathbb{R}^D$ in a large collection $\mathcal{Y} \subset \mathbb{R}^D$ of vectors. The neighborhood $\mathcal{N}(x)$ of the query vector x should reflect the objects that are close enough for a certain task. Formally, the collection \mathcal{Y} is a subset of a metric space (\mathbb{E}, d) . Two definitions of a neighborhood are given below.

- The ε -neighborhood is the set of vectors within a given distance to the query:

$$\mathcal{N}_\varepsilon(x) = \{y \in \mathcal{Y} : d(x, y) < \varepsilon\}. \quad (5.1)$$

- The k -nearest neighbors is the set of k vectors of \mathcal{Y} closest to x :

$$\mathcal{N}_k(x) = \{y \in \mathcal{Y} : y = k \arg \min_{y \in \mathcal{Y}} d(x, y)\}. \quad (5.2)$$

Other kinds of neighborhoods are considered to overcome the limitations of these definitions. For instance, reciprocal neighborhoods and adaptive neighborhoods [Jégou et al. 2010, Danfeng et al. 2011, Delvinioti et al. 2014] have been proposed to avoid *hubs* [Radovanović et al. 2010]. Other strategies [Lowe 2004, Omercevic et al. 2007, Joly and Buisson 2009, Rabin et al. 2008, Rabin et al. 2009, Furon and Jégou 2013] determine how meaningful the neighbors are.

The Euclidean case (\mathbb{R}^D, ℓ_2) is of particular interest, as it is both intuitive and compatible with numerous mathematical tools, in particular linear algebra. Sim-

ilarly, the cosine similarity is related to the Euclidean distance through the polarization identity $d(x, y)^2 = \|x - y\|^2 = \|x\|^2 + \|y\|^2 - 2\langle x|y\rangle$; therefore the neighborhoods it defines are identical for normalized vectors. Most of this chapter is devoted to Euclidean and cosine metrics, except Section 5.6, which specifically discusses non-Euclidean metrics.

5.1.1 Exact, Exhaustive, and Approximate Search Algorithms

Numerous multi-dimensional indexing methods have been proposed to reduce search time, such as the popular KD-tree [Friedman et al. 1977] or other branch and bound or partitioning techniques (see Samet [2007] for an overview). These methods organize the set of vectors \mathcal{Y} offline in an *index*, in order to reduce the number of computations at query time. Some of these structures can be thought of as the multi-dimensional counterparts of the indexing methods designed for the one-dimensional case. For instance, the R-tree [Guttman 1984] shares several properties, such as dynamic balancing, with the B-tree [Bayer and McCreight 1970], which is used in relational databases or for file systems such as NTFS or Ext4. Note that for the case $D = 1$, nearest-neighbor (NN) search is performed in worst-case logarithmic time with no significant memory overhead.

However, early multi-dimensional indexing structures were designed for low-dimensional vectors such as the ones used in geo-data applications. In contrast, high-dimensional data is prevalent in visual search, with descriptors such as the scale invariant feature transform (SIFT) [Lowe 2004] or features extracted with a convolutional neural network [Babenko et al. 2014]. In high dimensions, it turns out [Weber et al. 1998] that exact approaches are no more efficient than the brute-force exhaustive distance calculation, whose complexity is linear in the cardinality $|\mathcal{Y}|$ of the vector set.

There is a large body of literature on algorithms that overcome this issue by performing ANN search. Instead of exactly computing the neighborhoods, the key idea shared by these algorithms is to find the neighbors with high probability “only.” These strategies are either derived from exact algorithms such as kd-tree, which can be explored with a best-bin-first strategy [Beis and Lowe 1997], or statistically inspired by the Johnson-Lindenstrauss Lemma [Johnson and Lindenstrauss 1984], like LSH [Indyk and Motwani 1998]. Such algorithms are well suited to multimedia search, where huge volumes of descriptors are required. Keeping in mind that image description and corresponding comparison metrics are imperfect anyway, in some situations it is possible to improve efficiency by orders of magnitude, without dramatically affecting overall search quality, by searching approximate neighbors instead of true ones.

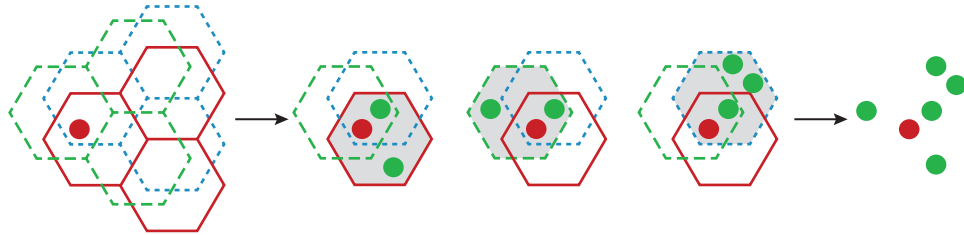


Figure 5.1 The cell-probe model, such as the one considered in [Datar et al. 2004]. Several partitions of the space are defined. Offline, each of the database vectors (in green) is associated with a cell for each of the partitions. At query time, the set of potential nearest neighbors is the union of the vectors sharing at least one cell with the query (in red).

Most exact and approximate nearest neighbors algorithms are described by the so-called *cell-probe model* [Yao 1981, Indyk and Motwani 1998, Miltersen et al. 1998, Andoni 2009]. In short, each indexed vector is assigned to one or several cells, as illustrated by Figure 5.1. At query time, a set of cells is also selected. The associated vectors are probed and returned as the set of potential nearest neighbors for the submitted query. In the case of exact search, the vector is guaranteed to belong to one of these cells, while ANN methods only provide probabilities (under some assumptions) or empirical evidence.

At this stage, it is important to make the distinction between exact and exhaustive search.

- An *exact* algorithm returns the solutions associated with the desired definition of the neighborhood, as defined for instance by Equations 5.1 or 5.2. It is not necessarily exhaustive and the purpose of many indexing structures is precisely to compute the distances $d(x, y)$ on a subset instead of the full set \mathcal{Y} . This is typically done by pruning entire regions of the feature space based on the triangular inequality, in order to guarantee the exactness of the search. As discussed above, such approaches are effective in low-dimensional spaces, yet they fail in high-dimensional spaces.
- An *exhaustive* algorithm compares the query vector with all the elements in \mathcal{Y} . Its complexity is linear by definition. Exhaustive algorithms are not necessarily exact, in particular if the metric $d(x, y)$ itself is approximated, as discussed later in this chapter.

5.1.2 Evaluation Criteria

Time and space complexity are two important properties of similarity search techniques. The search accuracy is additionally required in the case of *approximate neighbor search*, as it trades the formal guarantee of finding exactly the desired neighbors against a large gain in space and time complexities. We consider the Euclidean case $\mathbb{E} = (\mathbb{R}^D, \ell_2)$ and a dataset \mathcal{Y} comprising $N = |\mathcal{Y}|$ vectors of dimension¹ D .

Time complexity. The efficiency of the search is generally meant *at query time*, as the time spent to prepare the index is usually less important, especially when the query must be returned to an impatient human user. Another reason is that the expected number of queries submitted to the system is assumed to be larger than the number of elements indexed by the system. This is the case for text queries on the web: in 2012, Google answered over 1,800 billion queries, while it indexed only an estimated 50 billion web-pages.

However, the index construction must remain tractable. As a counter-example, it is computationally infeasible to pre-compute the Voronoi diagram of a large set of high-dimensional vectors, as effective a strategy as it is for the 2-dimensional case (see, *e.g.*, [de Berg et al. 2008]). Similarly, the off-line computation of the k-NN graph is often regarded as being prohibitive for large values of D and N , although some recent works have shown it feasible on millions [Dong et al. 2011] or even hundred millions of vectors [Douze et al. 2016].

There are several ways to define the computational cost. The first important factor is the complexity exponent of the number of operations, which gives the asymptotic behavior of the algorithm as the database size grows to infinity. However, Weber et al. [1998] show the limitations of this complexity measure for exact search. They conclude (in Section 3.4) that all clustering or partitioning techniques tend to a linear complexity $\mathcal{O}(N)$ when D is large enough. This observation motivates the introduction of the vector approximation file (VA-file) [Weber and Blott 1997, Weber et al. 1998]. This strategy, albeit exhaustive, reduces the number of operations by performing fewer operations for most of the vectors, thanks to a

1. By dimension, we here refer to the outer dimension of the vectors. The complexity of some algorithms is better related to the intrinsic dimensionality of the vectors, for instance if they belong to a linear subspace. However, we will omit this distinction in the following, as it goes beyond the scope of this manuscript.

compressed representation of these vectors. This structure can be seen as an ancestor of the quantization-based techniques presented in Section 5.4.

From a more pragmatic point of view, one may measure the complexity of a similarity search in a given application scenario, that is, for a particular kind of vectors. In this case, the parameter D is part of the problem and the goal is to measure the complexity as a function of the number of vectors N . We consider three metrics for measuring the efficiency.

The number of operations, in particular distance computations, is a reasonable measure for techniques based on clustering or, more generally, space partitioning. It is typically employed with the cell-probe model. It is decomposed [Paulevé et al. 2010] into:

- The *query preparation cost*, which does not depend on N . It is, for instance, the complexity to compute a hash code associated with the query vector.
- *Selectivity*, which is the rate of vectors marked as potential neighbors by the algorithm.

Considering an algorithm compatible with the cell-probe model depicted in Figure 5.1, this complexity includes (i) selecting the cells (or regions) to be checked, and (ii) computing the true distances, which, typically, linearly depends on the selectivity. For arbitrarily large datasets, the complexity only depends on the selectivity, which gives the fraction of distance computed over the exhaustive case. Counting the number of operations has several limitations when estimating the concrete efficiency of a method. In particular, the number of operations assumes a simplistic complexity model. Look-up operations are not properly modeled. For instance, random access to memory is assumed to be as fast as a sequential scan. It does not consider the specialized processor operations (e.g., as provided by SSEx instruction sets).

The amount of memory read is an interesting complexity measure for algorithms such as the VA-file [Weber and Blott 1997, Weber et al. 1998], sketches [Charikar 2002, Dong et al. 2008a], spectral hashing [Weiss et al. 2009], or product quantization [Jégou et al. 2011], as it determines the amount of computations performed during the retrieval phase. For large datasets, accessing the main memory, even in sequential order, may be the bottleneck: Reading a vector from memory takes more time than the actual distance computation involving the vector. This is often the case on modern server architectures.

The memory frequency is lower than that of the CPU and the memory bus is shared by several cores for multi-core processors. In contrast, integrated instructions now speed up the computation of inner products, Euclidean or binary distances.

The average query time is also a popular way to measure the complexity of ANN algorithms. It has the advantage of better reflecting the performance on complex architectures without requiring an explicit and complicated complexity model. As such, it can compare techniques with different mechanisms, such as methods using large amounts of disk, against techniques operating in main memory. This measure also suffers several drawbacks. First, timings' lack of reproducibility: they are not comparable on different machines, as many factors affect the overall speed, such as the processor speed, the amount of cached memory, and the speed of the memory bus. On the same machine, the operating system and more generally the installed software have a significant impact. For instance, for matrix multiplication, there can be an order of magnitude in speed between different versions of basic linear algebra subprograms (BLAS) [Dongarra et al. 1990]. In addition, it is difficult to compare different algorithms in the literature, as timings strongly depend on the implementation and because programming with the same (best possible) level of optimization is difficult. Finally, the timings may be polluted by concurrent programs executed by the system or users. For these reasons, conclusions drawn on a particular architecture or machine may not necessarily translate to another. Even when the experiments are carefully carried out, timings should be interpreted as indicative.

Space complexity. In practical applications, space complexity is an important criterion. Any improvement on this point makes it possible to index more vectors with the same resources. Another and probably more important reason for taking this criterion seriously is due to the memory hierarchy: An algorithm employing less memory is executed with faster memory. There are orders of magnitude in speed between processor caches, main memory, and hard-drive (either SSD or mechanical). Similarly, an index taking a few gigabytes of memory is amenable to being stored in the main memory of GPU cards [Wieschollek et al. 2016], which can carry out massive parallel computation instead of relying on the CPU. In most of the algorithms employed in visual search, memory is linearly related to the number of indexed vectors once the parameters are set for the feature distribution. In this document, we consider the number of bytes required per indexed vector. We distinguish three components requiring memory:

Storing the vector identifiers, although an identifier only requires $\lceil \log_2 N \rceil$ bits (*e.g.*, less than 4 bytes for 1 billion vectors), algorithms such as E²LSH [Andoni and Indyk 2006] need to store multiple identifiers. In contrast, this information is not required for a linear scan, as the indexing information is ordered consistently with the vectors.

The overhead of the indexing structure includes any meta-information required to perform a search, such as the overhead of hash tables or tree structure.

The raw vectors are required by some algorithms to perform a post-verification, or to update an initial ranking. They may be stored in main memory or on secondary mass storage.

Search quality. The search quality is measured with respect to the exact neighborhood of the query. A way to do it is to measure the proportion of ground-truth neighbors that are effectively returned. This measure must be balanced by another quantity limiting the number of returned neighbors. For instance, one may consider evaluating the following trade-offs:

- **selectivity vs recall.** The selectivity is related to the proportion of neighbors returned by the system. These vectors typically form the short-list to be post-processed. The average short-list size is obtained as $N \times \text{selectivity}$.
- **1-recall@ r** (or simply **recall@ r**) is the proportion of queries for which the nearest neighbor is ranked in the top r positions. This measure applies only when the algorithm returned an ordered list of neighbors, such as sketches [Charikar 2002], spectral hashing [Weiss et al. 2009], or product quantization [Jégou et al. 2011]. In contrast, it is therefore not adapted to partitioning-based techniques such as LSH, whose quality is better measured by the trade-off selectivity/recall.

Another popular way to measure the quality is the mean average precision. However, when the task is to find the nearest neighbor, it amounts to weighting the rank r of the true nearest neighbor by $1/r$. This choice is questionable from an application point of view, as a simple post-verification on the first vectors of a short-list dramatically increases this measure.

Other criteria. Depending on the application, several complementary characteristics may be required, such as dynamic insertion or deletion of vectors. Similarly, some algorithms assume that the bottleneck is the access time, for instance when working with mechanical drives [Lejsek et al. 2009] or in systems distributed on a

network [Moise et al. 2013]. Under this assumption, time efficiency is better modeled by the number of accesses, which reflects the number of moves done by the heads of a mechanical drive or the number of packet transmissions on a network. Another popular choice for the memory model, inherited from the database literature, is to count the number of disk cache misses induced by the algorithm. Finally, some metrics that are specific to binary codes have also been considered [Wang et al. 2012a], such as the precision at a given Hamming distance. However, such a metric has drawbacks: (i) it compares different methods at different recall/precision points, which does not allow a direct comparison of different binary methods as a function of the code length; (ii) it does not allow a direct comparison with non-binary methods.

5.2 Cell-probe Algorithms

Historically, the main search mechanism considered in the literature is the cell-probe model [Yao 1981, Gionis et al. 1999, Datar et al. 2004, Andoni 2009], depicted in Figure 5.1. In Euclidean variants of LSH, a cell is implicitly defined by a hash function g of the form

$$g : \mathbb{R}^D \rightarrow \mathbb{F} \quad (5.3)$$

$$x \mapsto g(x) \quad (5.4)$$

where \mathbb{F} is the discrete space of possible hash values. The cell associated with a given hash value g_v is therefore defined as $g^{-1}(g_v) = \{x \in \mathbb{R}^D : g(x) = g_v\}$. \mathbb{F} may be a finite space, but this is not strictly required. If the set of possible indexes is not finite but countable, as with unbounded lattice quantizers, they are hashed to a finite set. A secondary hash-key can be used to reduce the resulting collisions to an arbitrary small probability.

The hash function should be designed such that two points x and y that are close to each other are likely to be hashed in the same cell with “high” probability. This is formalized by the concept of locality-sensitive hashing functions, which satisfy

$$\mathbb{P}(g(x) = g(y)) \geq p_1 \text{ if } d(x, y) \leq d_1, \quad (5.5)$$

and, conversely, two far points are hashed differently:

$$\mathbb{P}(g(x) = g(y)) \leq p_2 \text{ if } d(x, y) \geq d_2, \quad (5.6)$$

where $d_1 \leq d_2$ and $p_1 > p_2$. A good hash function is such that $p_1 \gg p_2$. It is possible to magnify the ratio p_1/p_2 by compounding several hash functions, as done

in E^2 LSH with random projections [Datar et al. 2004]. These definitions of locality-sensitive functions are not directly related to the objective of finding nearest neighbors, as defined by Equation 5.1, but of *near neighbors*, which is closer to a range search criterion as defined in Equation 5.2. Note that LSH offers guarantees for the near-neighbor problem, not for the k-NN problem.

The key of LSH is to define multiple hash functions to achieve this guarantee. Therefore LSH relies on a set of L hash functions $\{g_j\}_{j=1\dots L}$, each inducing a partition of the feature space. Conceptually, LSH represents a vector as

$$G : \mathbb{R} \rightarrow \mathbb{F}^L \quad (5.7)$$

$$x \mapsto G(x) = [g_1(x), \dots, g_L(x)]. \quad (5.8)$$

At query time, one cell is probed from each partition. The set of vectors associated with these cells are considered as potential nearest neighbors: all vectors y such that $d_h(G(x), G(y)) < L$ are selected, where $d_h(\cdot, \cdot)$ represents the Hamming distance over \mathbb{F}^L . If the hash functions are independent, the probability of missing a near neighbor is bounded by $(1 - p_1)^L$ instead of $1 - p_1$ with a single hash function.

Remark 5.1 As there is no ranking between the retrieved vectors,² the distances between the query and all retrieved hypotheses are computed to produce a ranking. However, this requires storing all indexed vectors, which, for large sets, is possible on secondary mass storage only. For this reason, some strategies simply consider the database vectors co-hashed with the query as neighbors, as suggested for instance by Sivic and Zisserman [2003].

5.2.1 Hash Functions Bestiary

The LSH hash functions (equivalently, partitions) are popularly defined by random projections [Datar et al. 2004]. However, it is worth mentioning that this choice is not a requirement of the framework.³ The first hash functions defined by LSH were defined by taking a subset of the vector components [Indyk and Motwani 1998]. This construction is still the best to date for the Hamming space. In the Euclidean case, random projections have received some particular attention [Datar et al. 2004].

2. It is possible to order the vectors based on the number of probed cells in which they appear, yet this strategy is not effective when the number of hash functions is small, as it produces many ties.

3. It is also not specific to LSH: Other search algorithms such as Omedrank [Fagin et al. 2003] or the NV-tree [Lejsek et al. 2009] also use random projections.

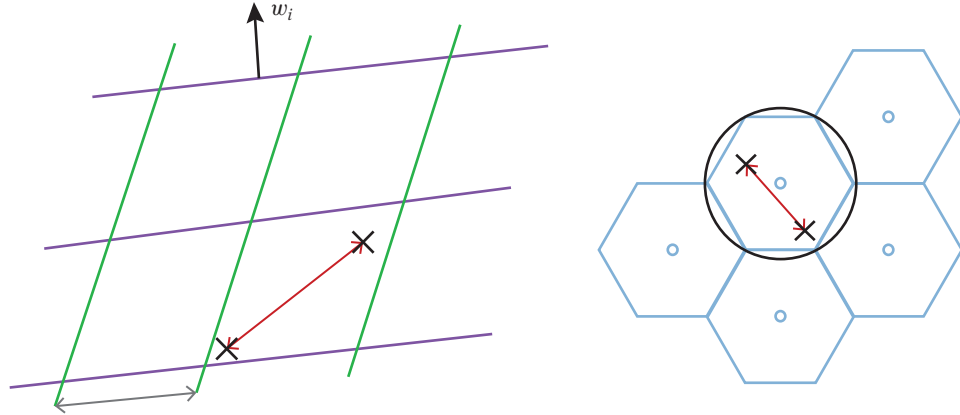


Figure 5.2 Example of two hash functions for LSH in the 2-D Euclidean case. *Left:* Random projection. *Right:* Hexagonal lattice. The elementary shape of the lattice is more compact, i.e., closer to a Euclidean ball. As a result, it is less likely that two far points are co-hashed in the same cell. Equivalently, a small perturbation on a given point is less likely to move the noisy point in another cell with the lattice.

Beyond random projections: structured and learned quantizers. More sophisticated quantizers such as lattice quantizers [Conway and Sloane 1982b] offer better quantization properties than random projection. This stems from the *shape gain*, the fact that the elementary cell in a good lattice is closer to the Euclidean ball than that possible with a separable distribution. This superiority also translates to a better performance for NN search under uniform distributions. In particular, the Leech lattice [Andoni and Indyk 2006] and the hyper-diamond E8 lattice [Jégou et al. 2008a] have been proposed with LSH and demonstrated better than random projections by [Andoni and Indyk 2006, Andoni 2009]. Figure 5.2 gives an intuition of why lattice quantizers offer better performance than random hyper-parallelotopes. Lattices (more generally, structured quantizers) are associated with efficient algorithms [Conway and Sloane 1982a, Vardy and Be’ery 1993, Agrell et al. 2002], enabling efficient quantization,⁴ comparable to if not faster than performing a projection.

Yet all these regular constructions are not adapted to the data, or are in a limited way by (automatic) parameter optimization [Dong et al. 2008b, Slaney et al. 2012].

4. The algorithms for encoding with the Leech lattice suggested in Andoni and Indyk [2006] are non-trivial. The hyper-diamond lattice E8 offers a faster hashing time while offering competing performance [Jégou et al. 2008a].

In contrast, learned quantizers such as kd-tree [Muja and Lowe 2009] or k-means [Paulevé et al. 2010] inherently adapt the cell size to the distribution of the features, but do not offer the same theoretical guarantee. In computer vision, the hash functions adapted to the feature distribution are generally more effective. These conclusions complement those of Philbin et al. [2008], who show that a bag-of-words representation defined by a fixed grid quantizer [Tuytelaars and Schmid 2007] leads to poor image retrieval performance. Paulevé et al. [2010] discuss and compare several families of hash functions.

The imbalance factor. The reason why hash functions learned on the data are more effective than structured ones, with respect to the trade-off between selectivity and recall, is that all cells induced by a structured hash function have equal sizes. If the probability distribution function of the vectors significantly deviates from uniformity, which is the typical case, the cells have uneven populations. This severely impacts the selectivity, which is minimized when all the cells contain the same population [Paulevé et al. 2010]: if K is the number of cells, the best possible selectivity is $1/K$.

This problem appears with the bag-of-words representations of images. In the seminal work by Sivic and Zisserman [2003], it was addressed by using *stop-words*, like in text-retrieval [Manning et al. 2008 (Chapter 2)]. This strategy simply amounts to removing the most populated cells. For nearest-neighbor search, it means that the query and database vectors that are hashed in one of these cells are not both considered. Nistér and Stewénius [2006] notice that this choice reduces the performance for image retrieval. This phenomenon can be measured by the so-called *imbalance factor* (IF). It was first empirically measured in image retrieval [Nistér and Stewénius 2006]. Yet it is possible to compute it [Jégou et al. 2010a] from the probability mass function associated with the cells. Let us denote by

$$p_i = \mathbb{P}(g(x) = \mathcal{C}_i) \quad (5.9)$$

the probability that a vector is assigned to cell \mathcal{C}_i . For a set comprising $N = |\mathcal{V}|$ vectors, the expected number of vectors lying in \mathcal{C}_i is $N \times p_i$. Assuming that the query follows the same distribution as the indexed vectors, the expected number of selected vectors, that is, the size of the short-list (denoted \mathcal{L}), is

$$\mathbb{E}_X[|\mathcal{L}|] = N \sum_{i=1}^K p_i^2. \quad (5.10)$$

This quantity is minimized when all p_i are equal, in which case $\mathbb{E}_X[|\mathcal{L}|] = N/K$.

The factor of sub-optimality resulting from the uneven distribution, termed *the imbalance factor* (IF), is obtained as

$$\text{IF} = K \sum_{i=1}^K p_i^2. \quad (5.11)$$

It is equal to 1 when $\forall i, p_i = 1/K$ and is larger otherwise. Note that a kd-tree achieves this lower bound thanks to the hierarchical median splitting construction, which may explain the good performance reported with multiple kd-trees for ANN search [Muja and Lowe 2009]. The selectivity of an algorithm using a single hash function g is given by

$$\text{selectivity} = \sum_{i=1}^K p_i^2. \quad (5.12)$$

Discussion 5.1 K-means clustering does *not* approximate the distribution of the vectors from which it is learned. This explains why the corresponding imbalance factor deviates significantly from 1. High-rate quantization theory⁵ [Panter and Dite 1951, Gray and Neuhoff 1998] establishes that for a scalar random variable X with probability distribution function p , the asymptotic distribution of centroids follows a distribution in $p^{1/3}$ in the scalar case [Bennett 1948]. Zador later extended the analysis to higher-dimensional space, also showing that the distribution of the centroids does not follow that of the original vectors. This confirms that the cells resulting from k-means partitioning for a non-uniform distribution have different populations.

Optimizing hash functions. LSH uses a pool of hash functions. There are two ways to improve this pool.

Optimizing individual hash functions. With a fixed quantizer such as random projections or a lattice, some parameters have to be fixed to adjust the size of the cells. For instance, the volume of the elementary lattice cell is adjusted by a scaling factor [Andoni and Indyk 2006, Jégou et al. 2008a]. Optimizing this parameter is critical in LSH. For this reason, several works propose strategies to automatically find good parameters [Dong et al. 2008b, Muja and Lowe 2014, Slaney et al. 2012], or even to select the most appropriate kind of hash functions [Muja and Lowe 2009, Muja and Lowe 2014].

Designing complementary hash functions. It is more difficult to predict the selectivity and recall resulting from the use of multiple hash functions. Some

5. The number of centroids tends to infinity and p is assumed locally constant.

vectors are retrieved several times, especially those that are close to the query. Two hash functions may be redundant, which is not desirable. The problem of finding complementary hash functions beyond the natural complementarity provided by random generation has recently received some attention [Xu et al. 2011, Liu et al. 2013b, Jin et al. 2013].

5.2.2 Query Mechanisms

In this subsection, we review two variants of LSH, within the probe model, that modify the query procedure in two opposite ways. The first *multi-probe* LSH is worth considering when the memory is a critical resource and/or the (fixed) hashing complexity is non-negligible with respect to the database side. The second, in contrast, is worth considering when databases are very large and available memory is unlimited. These methods are compatible, with limited adaptation, with most kinds of hash functions, as the ones discussed above.

Multi-probe. LSH as a hashing algorithm (i.e., associated with the probe model) has a major drawback: It is inefficient from a memory point of view, for two reasons:

- Raw vectors are required for post-verification.⁶ This issue is solved, at least partially, by the hybrid algorithms introduced in Section 5.5.
- Multiple partitions—due to the partitioning nature of the algorithm, L hash tables are constructed, each introducing some memory overhead: A vector identifier must be stored for each of the L hash functions. Considering for instance $L = 8$ and a database comprising $N = 1$ billion vectors, it means that it takes about 32 GB to store the vector identifiers.

The multi-probe LSH [Lv et al. 2007] addresses this issue by considering a single hash function (or a few), but retrieves several cells instead of one per hash function, as illustrated in Figure 5.3. In addition to the cell to which the query is normally associated, additional cells are probed based on their likelihoods to contain neighbors. This is done by considering a perturbation vector, initially applied directly on the hash keys. Joly and Buisson [2008] improve the cell selection model for LSH based on random projections. Paulevé et al. [2010] propose a selection rule for k -means based on k nearest centroids. Multi-probe LSH is, to some extent, similar to older strategies with prioritized search, such as the best-bin-first variant of the kd-tree [Beis and Lowe 1997].

6. Except if all retrieved hypotheses are assumed neighbors. Although this strategy is sub-optimal (it returns many outliers), it may be sufficient in situations where subsequent filtering rules are employed, such as spatial verification.

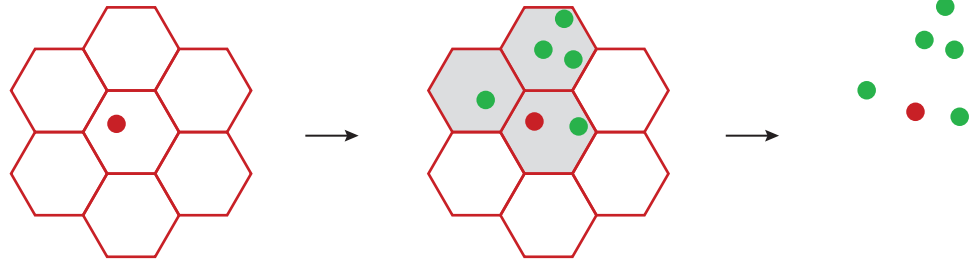


Figure 5.3 Multi-probe LSH. For a given partition, several cells are probed based on a relevance criterion.

Multi-probe LSH is a bit inferior to regular LSH with respect to the selectivity/recall trade-off [Paulevé et al. 2010]. Yet this is an effective strategy to reduce the large memory requirements of LSH. Another advantage worth mentioning is that each vector is retrieved once at most, thus eliminating the need extract to the unique elements from the short-list.

Remark 5.2 The multi-probe variant of LSH is closely related to the multiple assignment strategy in bag-of-words [Jégou et al. 2007, Philbin et al. 2008], especially when considering the asymmetric assignment to visual words [Jégou et al. 2010a].

Query-adaptive LSH. For very large datasets, complexity is best reflected by the selectivity, as discussed in Section 5.1.2. The query-adaptive strategy introduced in Jégou et al. [2008a] aims at optimizing the trade-off between selectivity and recall, regardless of memory usage and the computational cost of identifying the cell.

It works as follows. A pool of hash functions is constructed. This pool is typically larger than in LSH. The first step determines the most relevant set of hash functions associated with the input query. As in multi-probe LSH, a relevance criterion is designed to rank hash functions (as opposed to cells for multi-probe) by decreasing order of expected relevance. In the second step, a single cell is probed for all the selected hash functions, i.e., those best ranked according to the relevance criterion, and none for the others.

Query-adaptive LSH was first proposed with random projections and the E8 lattice [Jégou et al. 2008a] before being extended to k-means partitioning [Paulevé et al. 2010]. For random projections, the criterion is defined by the distance (after projection) to the closest boundaries of the partition. For lattices and k-means, the distance to the closest centroid in each partition is used instead.

Query-adaptive LSH is empirically evaluated in Paulevé et al. [2010]. This strategy always improves the trade-off between selectivity and recall. However, the fixed

query preparation cost (i.e., computing the relevance criterion) is increased by this strategy. It is therefore better associated with a hash function requiring few computations, like structured lattice quantizers. In addition, memory should be available in large quantities: The larger the pool of hash functions, the better is the gain with respect to the selectivity/recall compromise.

5.3 Sketches and Binary Embeddings

In most papers, LSH is not considered in the context of probe algorithms, but employed with a distinct comparison strategy [Lv et al. 2004], often referred to as *sketches*. Some approximate algorithms are regarded as the ancestors of these sketches. Two examples are famous: 1) the paper by Flajolet and Martin [1985] estimates the number of unique elements in a very large database without sorting the elements nor employing hash tables, and with little extra storage; 2) the min-hash algorithm by Broder [1997], which estimates the Jaccard similarity between sets.

In this section, we focus on sketches to address the main limitation of the probe model, i.e., the storage overhead. ANN probing algorithms are mainly compared based on the trade-off between search quality and efficiency, the memory requirements of the indexing structure being considered as a secondary criterion. For instance, the storage overhead of E^2 LSH is typically comparable to that of the original vectors. Although this problem is partly solved by the multi-probe variant (see Section 5.2.2), the recall is reduced for the same selectivity. Additionally, a final re-ranking step based on exact ℓ_2 distances is required to get a reliable ranking, which limits the number of vectors that can be handled by these algorithms.

For this reason, we have witnessed a renewed interest for techniques considering the memory usage as a primary criterion in a context of visual search. The application of sketches to image retrieval with compact signature was pioneered by computer scientists [Indyk and Thaper 2003, Lv et al. 2004] shortly after the cosine sketches by Charikar [2002].

Binary sketches are the most popular. In this context, sketches amount to replacing the original descriptors (typically Euclidean) by short binary codes, as shown in Figure 5.4. They are either integrated directly in the descriptor construction [Naturel and Gros 2008, Strecha et al. 2012] or considered as part of the indexing strategy of local [Jégou et al. 2008b] or global descriptors [Lv et al. 2004, Torralba et al. 2008].

The literature on these binary sketches is extremely vast, with many variants considered. For an overview on this subdomain, we refer the reader to two recent surveys on binary hashing, by [Wang et al. 2014] and [Wang et al. 2016a]. As a side

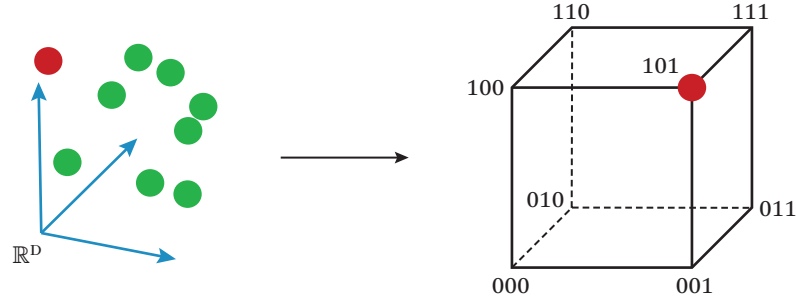


Figure 5.4 Binary sketches cast the Euclidean nearest neighbor problem into the Hamming space.

note, it was shown in Sablayrolles et al. [2016] that many recent papers proposing to learn binary codes in a supervised manner have used a flawed evaluation protocol. They are outperformed by a trivial baseline encoding the output of a classifier. If we exclude these inconclusive works, one of the most successful scheme, in this context is designed with quantization [He et al. 2013].

5.3.1 Similarity Estimation with Sketches

To the best of our knowledge, it is Charikar [2002] who first introduced sketches for fast estimation of cosine similarity in the Hamming space. Focusing on the case of cosine, the method is based on random projections. Each projection is defined by a vector w_j , $j = 1 \dots L$. For any vector $x \in \mathbb{R}^D$, each projection produces a bit

$$b_j(x) = \text{sign}(w_j^\top x). \quad (5.13)$$

The sketch of x consists of the concatenation of these bits in a binary vector:

$$b(x) = [b_1(x), \dots, b_L(x)]. \quad (5.14)$$

Remark 5.3 This measurement strategy is called “1-bit compressive sensing” [Boufounos and Baraniuk 2008] in the signal processing community. It is also closely related, as a particular case, to early papers on quantized frame expansion [Goyal et al. 1998].

Let us assume that the projection directions W_j are random variables independently and uniformly drawn on the unit sphere. Now consider the subspace spanned by two unit-norm vectors x and y . The intersection of this subspace with the hyper-plane defined by W_j is a 1-dimensional vector subspace. Because all directions are equivalent, the probability that it separates x from y is directly related to the absolute value of (unsigned) angle θ between x and y , such that $\cos \theta = \langle x|y \rangle$.

This gives us the following key property:

$$\mathbb{P}(b_j(x) = b_j(y)) = 1 - \frac{1}{\pi} \arccos \langle x|y \rangle. \quad (5.15)$$

Since we consider bits compared with Hamming distance, we also have

$$\mathbb{P}(b_j(x) = b_j(y)) = 1 - \mathbb{E}_{W_j}(d_h(b_j(x), b_j(y))). \quad (5.16)$$

Thanks to the linearity of the expectation and the fact that the W_j are drawn independently one from each other, the Hamming distance gives an unbiased estimator of the cosine similarity:

$$d_h(b(x), b(y)) \approx \frac{L}{\pi} \arccos \langle x|y \rangle. \quad (5.17)$$

In high dimensional spaces, two vectors have an angle close to $\theta = \pi/2$ with high probability. Considering the Taylor expansion of arccos in 0, $\arccos u = \pi/2 - u + \mathcal{O}(u^3)$, we have

$$\frac{L}{\pi} \arccos \langle x|y \rangle = \frac{L}{2} - \frac{L}{\pi} \langle x|y \rangle + \mathcal{O}(\langle x|y \rangle^3) \quad (5.18)$$

and evidences that the inner product is well approximated close to 0 by the Hamming distance, as

$$\langle x|y \rangle \approx \frac{\pi}{L} d_h(b(x), b(y)) - \frac{\pi}{2}. \quad (5.19)$$

After this seminal work by Charikar, several researchers have proposed other kernel estimations from binary codes [Rahimi and Recht 2007, Weiss et al. 2009, Raginsky and Lazebnik 2010, Gong and Lazebnik 2011]. It is worth noting that similar popular sketch constructions have been introduced in different communities: For instance, spectral hashing [Weiss et al. 2009] is close to the ℓ_2 binary sketches of Dong et al. [2008a]. These constructions are designed for the ℓ_2 distance and share the idea of cyclic quantization depicted in Figure 5.5. The same idea is considered in a compressive sensing framework [Boufounos 2012].

Link between sketches and multi-probe LSH. Binary sketches are compared with the Hamming distance. Typically, a threshold h_t is defined and two vectors are deemed neighbors if $d_h(b(x), b(y)) \leq h_t$. Recall that the multi-probe variant of LSH assumes a perturbation vector. In the case of binary sketches, this perturbation vector can be defined by flipping bits ($0 \rightarrow 1$ or $1 \rightarrow 0$). By bounding the number of flipped bits by h_t , we obtain the same neighbors as those defined by sketches.

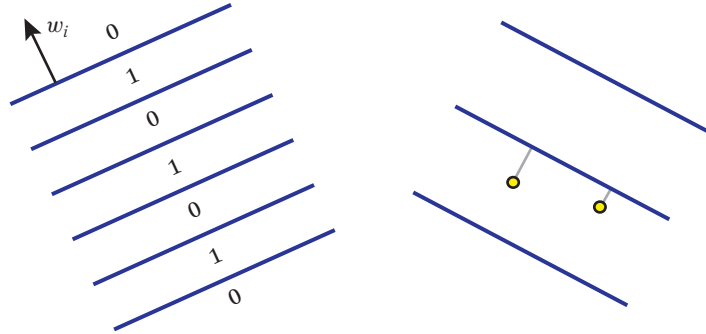


Figure 5.5 Illustration of the ℓ_2 -sketch of Dong et al. [2008a]. *Left*: The construction of a bit in a sketch alternates regions assigned to 0 and 1. *Right*: A better distance estimator is obtained by computing an asymmetric distance involving the distance between the query and its closest hyper-plane.

Asymmetric scheme with sketches. One of the main features of sketches is their compactness. In an image retrieval scenario, they represent an image with few bits, thereby making it possible to store millions to billions of image descriptors in memory. However, the memory constraint is not necessarily required for the query, as this one is processed online and its descriptor may be discarded afterward.

This observation motivates the use of asymmetric methods, in which database vectors are encoded with short codes but the query is kept uncompressed to avoid quantization errors. This scheme was first proposed with sketches [Dong et al. 2008a], where the authors estimated an asymmetric distance based on distances to the separating hyper-plane, as illustrated by Figure 5.5. It is extended to quantization-based techniques [Sandhawalia and Jégou 2010, Jégou et al. 2011] to produce a better distance estimate, as discussed in Section 5.4.

5.3.2 Hash Function Design

Similar to LSH with a probe mechanism, the performance of sketches depends on the design of the hash functions. The random projections proposed by Charikar [2002] are standard for the cosine similarity. However, by adopting a reconstruction point of view, they appear [Balu et al. 2014] to be suboptimal. For the cosine similarity, we consider two cases:

- $L \leq D$. Using a set of orthogonal vectors yields better results than random projections [Jégou et al. 2012], as first proposed in Jégou et al. [2008b]. Note that the methods performing a principal component analysis (PCA) rotation,

such as spectral hashing [Weiss et al. 2009], implicitly use a set of orthogonal vectors.

- $L > D$. When the number of bits produced is larger than the dimensionality of the input feature, it is no longer possible to generate a subset of L orthogonal features. The L projection vectors form an over-complete frame $W = [w_1, \dots, w_L]$ [Goyal et al. 1998]. Inspired by results on quantized frame expansion, Jégou et al. [2012] propose to use a tight uniform frame (or Parseval frame)—i.e., such that $W.W^T = I_D$ —instead of random projections. The superiority of this strategy is independently confirmed by Simonyan et al. [2013]. Another concurrent strategy [Ji et al. 2012] combines subsets of orthogonal vectors (called *super-bits*). The authors mention that the best results when $L \leq D$ are achieved with a complete basis.

Other strategies optimize the embeddings [Kulis and Darrell 2009] in order to better reconstruct the small distances. Another popular approach is to directly optimize a rotation in order to balance the variance on the different components [Jégou et al. 2010b, Gong and Lazebnik 2011], such that each bit gives the same approximation error. These works mainly differ by the parametrization, for instance a Householder decomposition in one case, and by the way the optimization is carried out. Therefore the quality of the estimated rotations should not differ significantly, although we are not aware of a direct comparison in the literature.

5.3.3 Beyond “Project and Binarize”

Formally, the “project+take sign” approach defines a spherical quantizer. It implicitly defines a set \mathcal{C} comprising at most 2^L distinct reproduction values (centroids) of the form

$$c \propto \sum_{j=1}^L a_j w_j, \quad (5.20)$$

where $a_j = \pm 1$. The proportionality constant is determined such that $\|c\| = 1$. We do not discuss possible degenerated cases such that $\sum_{j=1}^L a_j w_j = 0$. An interesting question is related to the binarization strategy proposed in Equation 5.13. Does the centroid c_i selected by this strategy provide the best possible choice from a reconstruction point of view? The best centroid is the one minimizing the ℓ_2 distance, which is equivalent for ℓ_2 -normalized vectors to maximizing the co-linearity to the input vector x :

$$c^* = \arg \max_{c \in \mathcal{C}} x^\top c = \arg \max_{c \in \mathcal{C}} \frac{\sum_{j=1}^L a_j x^\top w_j}{\|\sum_{j=1}^L a_j w_j\|}. \quad (5.21)$$

Let us first consider the case of an orthonormal set of vectors: $\forall j, j', w_j^\top w_{j'} = 0$ and $\forall j, \|w_j\| = 1$. In this case, the denominator is constant and the optimum is obtained when, for $j = 1 \dots L$, a_j and $x^\top w_j$ have the same sign. But this property does not necessarily hold when the frame vectors are not orthogonal, meaning that a better reconstruction, *not* taking the sign of $x^\top w_j$, is possible.

Example 5.1 Let us consider the frame

$$W = [w_1 w_2 w_3] = \begin{bmatrix} 1 & 0 & \cos \frac{\pi}{3} \\ 0 & 1 & \sin \frac{\pi}{3} \end{bmatrix} \quad (5.22)$$

and define a matrix A for all possible bit combinations in Equation 5.20:

$$A = \begin{bmatrix} -1 & -1 & -1 & -1 & +1 & +1 & +1 & +1 \\ -1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 \\ -1 & +1 & -1 & +1 & -1 & +1 & -1 & +1 \end{bmatrix}. \quad (5.23)$$

This frame implicitly defines the spherical quantizer, which we write in matrix form as

$$\begin{aligned} C &= [c_1 \dots c_8] \\ &= \begin{bmatrix} -0.626 & -0.966 & -0.996 & -0.259 & +0.259 & +0.996 & +0.966 & +0.626 \\ -0.779 & -0.259 & +0.089 & +0.966 & -0.966 & -0.089 & +0.259 & +0.779 \end{bmatrix}. \end{aligned} \quad (5.24)$$

Now, if we encode these centroids with Equation 5.20, we obtain

$$\text{sign}(W^\top \times C) = \begin{bmatrix} -1 & -1 & -1 & -1 & +1 & +1 & +1 & +1 \\ -1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 \\ -1 & -1 & -1 & +1 & -1 & +1 & +1 & +1 \end{bmatrix} \neq A,$$

which is obviously the best possible choice for the binary code.

In other terms, although the centroids in Equation 5.25 are perfectly reconstructible by selecting the proper binary codes, the strategy “project+take sign” takes suboptimal choices for two of them. Another way to see this issue is to observe that function $x \mapsto b(x)$ is not necessarily surjective, that is, there is a loss of capacity: some binary codes are never selected, and most binary codes are unlikely to be selected if $D < L$.

Anti-sparse coding and quantization-optimized LSH. These observations have motivated some proposals for better *encoding* strategies. A first approach based on spread representations [Fuchs 2011] assumes that the set of projections is already defined. The goal is then to reduce the quantization error underlying the binarization by the sign function. Since there is an optimal strategy (orthogonalization of W) for the case $L \leq D$, we focus on the case $L > D$. A direct minimization of the reconstruction error is not tractable for this discrete problem. Instead, our first proposal was to resort to an “anti-sparse coding” strategy. Given an input vector x to be encoded, it relaxes the optimization problem by minimizing the objective function

$$v^* = \min_{Wv=x} \|v\|_\infty. \quad (5.26)$$

This equation is similar in spirit to what is considered in sparse coding, except that the ℓ_0 (or ℓ_1) norms are replaced by ℓ_∞ . As a result, instead of concentrating the signal representation on a few components, anti-sparse coding has an opposite effect: It tends to spread the signal over all components, whose magnitude is comparatively less informative.

Interestingly, $L - D + 1$ components of v are provably stuck to the limit, i.e., equal to $\pm\|v\|_\infty$. As a result, the vector v can be seen as a *pre-binarized version* of the binary code that we want to produce. This is shown by Figure 5.6, which compares the output v^* obtained by the spread representation to the usual projection values $W^T x$. The subsequent binarization to $\text{sign}(v^*)$ introduces less quantization loss with our approach.

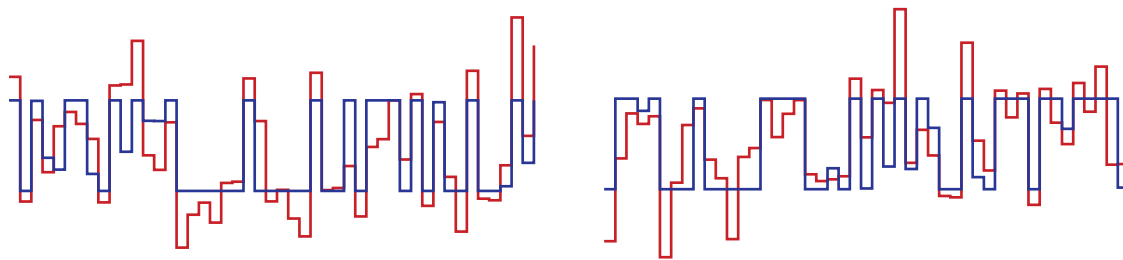


Figure 5.6 Illustration of an encoding method not based on the “project+sign” paradigm, namely anti-sparse coding. Compare the spread representation (in blue) with the values obtained by projecting the same input vector onto the frame (red). With the non-linear strategy, $L - D + 1$ components do not suffer any (spherical) quantization loss. This also translates to a better ranking compared with the “project+sign” method when employing the produced binary codes as sketches for NN retrieval.

Remark 5.4 Although this proposal achieves some gain over the “project+binarization,” it is worth mentioning that it does not guarantee finding the optimal spherical centroid with respect to the quantization error. Further gain is achievable, as shown by subsequent works such as the quantization-optimized LSH [Balu et al. 2014], which offers a more efficient encoding strategy than the optimization of Equation 5.26.

To conclude on binary sketches, significant gains are possible beyond the initial approach of Moses Charikar even without considering any kind of data adaptation:

- using an orthogonal set of vectors (if $L \leq D$) or a tight over-complete frame (if $L > D$);
- using an asymmetric scheme;
- using a better encoding approach to produce the binary codes, like anti-sparse coding;
- performing the explicit reconstruction of the vector (when the norms of the vectors are constant) in a re-ranking stage [Jégou et al. 2012].

5.4 Searching and Similarity Estimation with Quantization

The sketches discussed in the previous section are binary. This design choice leads to efficient implementations, thanks to low-level processor instructions, but drastically limits the quality of the quantizer underlying these sketches. For this reason, more general sketches providing better distance estimates have been proposed. Their motivation and intuition are that a better quantizer gives a better distance estimation, as suggested by a result [Jégou et al. 2011] that shows that the square Euclidean distance is statistically bounded by the mean square quantizer error if the quantizer satisfies Lloyd’s properties [Gray and Neuhoff 1998].

However, not all quantizers are suited to efficient distance estimation. Indeed, another property that the quantizer should offer is *compressed-domain distance estimation*. This requirement disqualifies quantizers such as the k-means, for which the quantization step becomes the bottleneck. Another reason why k-means is not a suitable option is that, in order to get good distance estimates, we need relatively long codes: It is not possible to learn a k-means with, for instance, 2^{128} centroids. Being able to quantize finely the vector space requires that the quantizer has some structure accelerating the assignment to the closest centroid. A contradictory requirement is that the quantizer should be adapted to the distribution to achieve good distance estimation, thereby excluding structured quantizers such as lattices.

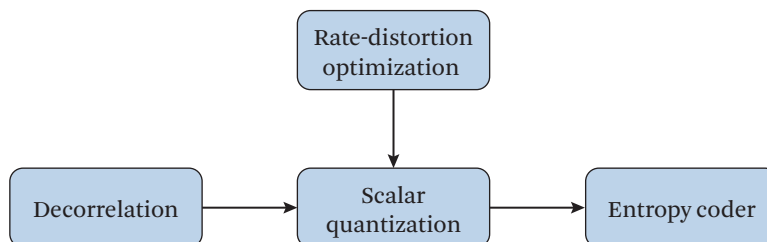


Figure 5.7 A typical source coding system as considered in our transform coding method.

5.4.1 Searching with Expectation

Two works [Sandhawalia and Jégou 2010, Brandt 2010] in this line of research mimic a traditional source coding (\approx compression) system, as the one depicted in Figure 5.7. After applying PCA (= de-correlation) on the input vector, a set of Lloyd-max scalar quantizers are optimized to provide the best average distance estimation. This is done by simply adjusting the number of centroids. A compounding strategy is proposed to accelerate the compressed-domain comparison.

5.4.2 Searching with Product Quantization

As already discussed in Section 5.2.1, vector quantization is better than scalar quantization thanks to the *shape gain* [Gray and Neuhoff 1998]. Formally, a quantizer relying on scalar quantization is suboptimal by construction because the elementary shape is a parallelotope, in the best case a hyper-rectangle if the projections used before quantizing are orthogonal. The shape of the corresponding cubic lattice \mathbb{Z}^D is significantly inferior with respect to different compactness criteria; see for instance Conway and Sloane [1990]. In addition, when doing vector quantization, more complex dependencies than simple correlation are likely to be exploited compared to a scheme like that of Figure 5.7. For these reasons, methods based on structure quantizers [Jégou et al. 2011] have been proposed to estimate (square) Euclidean distances from compact codes.

Distance approximation with quantized vectors. Let $x \in \mathbb{R}^D$ be our query vector and $\mathcal{Y} = \{y_1, \dots, y_N\}$ a set of vectors in which we want to find the nearest neighbor $\text{NN}(x)$ of x . The approach consists in encoding each vector y_i by a quantized version $c_i = q(y_i) \in \mathbb{R}^D$. For a quantizer $q(\cdot)$ with K centroids, the vector is encoded by $b_c = \log_2(K)$ bits, assuming K is a power of 2. An approximate distance $d_c(x, y_i)$ between a query x and a database vector is computed as

$$d_c(x, y_i)^2 = \|x - q(y_i)\|^2. \quad (5.27)$$

The approximate nearest neighbor $\text{NN}_a(x)$ of x is obtained by minimizing this distance estimator:

$$\text{NN}_a(x) = \arg \min_{y_i \in \mathcal{Y}} d_c(x, y_i)^2 = \arg \min_{y_i} \|x - q(y_i)\|^2, \quad (5.28)$$

which is an approximation of the exact distance calculation

$$\text{NN}(x) = \arg \min_{y_i \in \mathcal{Y}} \|x - y_i\|^2. \quad (5.29)$$

Note that we assume here that we use an asymmetric distance computation: the query x is not converted to a code. Therefore there is no approximation error on the query side. We also mention that this distance estimator is biased [Jégou et al. 2011] and show that this bias can be corrected. Empirically, this correction does not improve the search quality.

Product quantizer. To get a good vector approximation, K should be large ($K = 2^{64}$ for a 64 bit code). For such large values of K , learning a K -means codebook is not tractable; neither is the assignment of the vectors to their nearest centroids. A product quantizer [Jégou et al. 2011] addresses this issue because it does not require explicitly enumerating the centroids. Consider a vector $y \in \mathbb{R}^D$ split into m subvectors $y^1, \dots, y^m \in \mathbb{R}^{D/m}$. A product quantizer is defined as a function

$$q(y) = (q^1(y^1), \dots, q^m(y^m)), \quad (5.30)$$

which maps the input vector y to a tuple of indices by separately quantizing the subvectors. Each individual quantizer $q^j(\cdot)$ has K_s reproduction values, learned by K -means. To limit the assignment complexity, $\mathcal{O}(m \times K_s)$, K_s is set to a small value (e.g., $K_s = 256$). However, the set of K centroids induced by the product quantizer $q(\cdot)$ is large, as $K = (K_s)^m$. Figure 5.8 depicts a product quantizer with $m = 4$ and $K_s = 8$. Note that a product quantizer is fully parametrized by the number of subvectors m and the total number of bits b_c per subquantizer.

The squared distance estimation of Equation 5.28 is decomposed as

$$d_c(x, y)^2 = \|x - q(y)\|^2 = \sum_{j=1, \dots, m} \|x^j - q^j(y^j)\|^2. \quad (5.31)$$

The squared distances in the sum are read from look-up tables. These tables are constructed on-the-fly for a given query, prior to the search in the set of quantization codes, from each subvector x^j and the k_s centroids associated with the corresponding quantizer q^j . The complexity of the table generation is $\mathcal{O}(D \times K_s)$. When $K_s \ll N$, this complexity is negligible compared to the summation complexity of $\mathcal{O}(D \times N)$ in Equation 5.28.

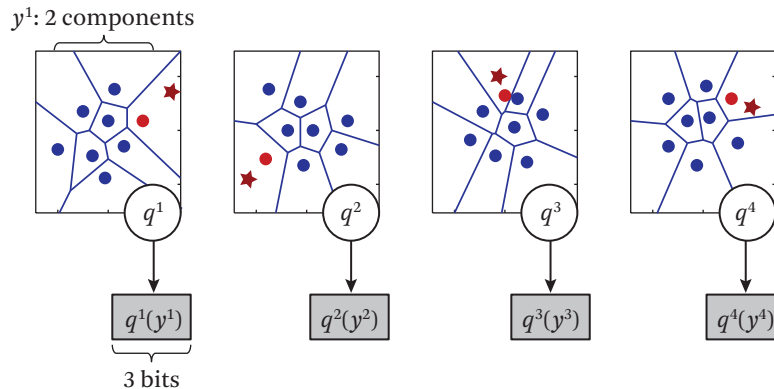


Figure 5.8 A simple product quantizer that splits an 8-D input vector into 4 subvectors, each quantized by a 2-D k-means. This quantizer allows the estimation of distances in the compressed domain.

Discussion. This approximate nearest neighbor method implicitly sees multi-dimensional indexing as a vector approximation problem: a database vector y is decomposed as

$$y = q(y) + r(y), \quad (5.32)$$

where $q(y)$ is the centroid associated with y and $r(y)$ the error vector resulting from the quantization, called the *residual* vector. Since the square error between the distance and its estimation is bounded, on average, by the quantization error (see [Jégou et al. \[2011\]](#)), asymptotically this scheme ensures returning perfect search results when increasing the number of bits allocated to the quantization indexes. Approaches based on product quantization codes define the state of the art in approximate search with compact codes and receive significant attention.

Note, however, that solely relying on product quantization codes is still an exhaustive approach requiring comparison of the query vectors to the codes of all database elements. Although this yields a constant acceleration factor and a significant memory savings, this is not sufficient to index billion-sized datasets. The hybrid approaches discussed later in [Section 5.5](#) propose a complementary way to improve the efficiency.

5.4.3 Subsequent Works on Product Quantization

Several works have extended this first encoding scheme. First, the coding quality is magnified by using a rotation of the input space and dimensionality reduction.

In particular, a PCA followed by a random or a learned rotation [Jégou et al. 2010b] can be performed before applying product quantization. Other strategies have been proposed to go beyond by jointly optimizing the rotation and the subquantizers, in particular optimized product quantization [Ge et al. 2013] and the Cartesian k -means [Norouzi and Fleet 2013].

Another work [Jégou et al. 2011] adopts a practical two-stage scheme in the spirit of partitioning techniques like E²LSH [Datar et al. 2004]. The main difference is that the second refinement stage does not rely on full vectors, but on an approximation that encodes the residual vector $r(y)$ in Equation 5.32. Doing so, it is still possible to maintain all the vectors in memory, while drastically improving the distance quality and therefore the ranking.

Other works [Babenko and Lempitsky 2014, Zhang et al. 2014, Zhang et al. 2015a] have pushed possible memory/efficiency trade-off by adopting a more general point of view, such as “Additive quantization” [Babenko and Lempitsky 2014], which provides an excellent approximation and search performance, yet is obtained with a much higher computational encoding complexity, leading the authors to subsequently propose more efficient encoding strategies [Babenko and Lempitsky 2015b]. In-between PQ and this general formulation, good trade-offs are achieved by residual quantizers, which can be used in the non-exhaustive PQ variant [Jégou et al. 2011] to reduce the quantization loss by encoding the residual error vector instead of the original vector; see next section. They can also be used as a coding strategy [Chen et al. 2010, Martinez et al. 2014, Ai et al. 2015]. An even more general model, based on quantized sparse coding, is proposed by Jain et al. [2016].

5.5 Hybrid Approaches: The Best of Probing and Sketching

In the previous sections, we have considered both probing algorithms and methods based on compact codes such as sketching or product quantization. Both mechanisms have advantages and drawbacks. An important observation that we make is that these two classes of approaches are not incompatible. What we call the hybrid approaches in this section are those which aim at giving the advantages of the two classes of techniques. This is done as follows:

- Adopt a probing algorithm at a coarse level, in order to filter out the vectors that are unlikely to be neighbors.
- Use a single hash function and a multi-probe strategy.
- For each of the probed vectors, exploit compact codes like binary sketches or quantized codes to give local measurements between vectors without requiring the original vectors.

The coarse hashing function is typically learned to adapt to the data distribution, as this gives better results [Paulevé et al. 2010]. In contrast, for the sake of efficiency comparison, one can rely on simpler hash functions for the local distance estimation, such as those associated with binary or quantized codes.

These hybrid strategies were initially designed as an extension of a bag-of-words approach in image retrieval [Jégou et al. 2008b], employing a k-means at the coarse level and binary codes obtained with an improved LSH binarization scheme. This approach, referred to as Hamming embedding, has been applied to local SIFT descriptors but also to global descriptors [Douze et al. 2009] like GIST [Oliva and Torralba 2001]. Later, the binary sketches by product quantization codes (see Section 5.4), and applies the global descriptors obtained by aggregating of local ones, such as VLAD features [Jégou et al. 2010b]. Doing this leads to a system able to find relevant images in a database comprising about 100 million images in about 200 milliseconds (on one core), with each image being represented by a short code. The timings are significantly improved with a GPU implementation [Johnson et al. 2017].

Alternative hybrid approaches have been recently proposed. One that received some particular attention is the inverted multi-index [Babenko and Lempitsky 2012], which employs product quantization for both the coarse partitioning and distance computation. This effective approach offers the advantage of accelerating learning and indexing, not only the query time.

Remark 5.5 The hybrid approaches normally give a good trade-off between search quality and efficiency. It is effective on most features, as long as their intrinsic dimensionality is not very large. With such approaches, it is possible to evaluate approximate search on billion-sized datasets, such as the BIGANN or Deep1B dataset. In the case of distributions with very high intrinsic dimensionality (uncommon with true data), it is better not to use any coarse level as the filtering stage is ineffective. In this case, one should rely on sketches only.

5.6 Searching for Non-Euclidean Metrics and Graph-based Approaches

Although most of the effort on ANN is devoted or demonstrated with the Euclidean distance, other metrics are of interest and various strategies have been proposed to tackle this case.

Several works revisit existing strategies to adapt them to new kernels. Note that the first LSH strategy was designed for the Hamming space [Gionis et al. 1999]. Some more specialized works intend to propose LSH for metrics like Hausdorff

metrics [Farach-Colton and Indyk 1999], Frechet distances [Indyk 2002] or Chi-Square [Gorisse et al. 2012]. A more generic strategy is kernelized LSH (KLSH) [Kulis and Grauman 2009], which extends LSH by employing the so-called kernel-trick; however, empirically this strategy is not very successful, as discussed below. In the same spirit, Joly and Buisson [2011] learn binary codes from support vector machine.

Another kind of strategy consists in casting non-Euclidean to Euclidean search. This strategy is advocated by Indyk and Naor [2007], who state that

“Combining the embeddings with known data structures yields the best-known approximate nearest-neighbor data structures for such metrics.”

Although some works rely on distance embeddings [Athitsos et al. 2008], a more direct strategy consists in using kernelized PCA jointly with a state-of-the-art Euclidean/cosine indexing scheme on the embedded vectors, or explicit feature maps like those proposed by Vedaldi and Zisserman [2012]. The strategy of combining an explicit embedding with a state-of-the-art indexing method like product quantization was reported [Bourrier et al. 2015] to offer better results than other works directly designed for a particular metric. These results concur with the conclusion of Indyk. As a byproduct of this evaluation, KLSH is also shown to be inferior to the simple combination of kernel PCA (KPCA) with the standard LSH binary sketches of Charikar [2002].

Similarly, for the specific and important case of the maximum inner product search, an explicit embedding strategy has been proposed [Shrivastava and Li 2014, Neyshabur and Srebro 2015] to take into account the norm of the indexed vectors.

Graph-based approaches. The NN-descent [Dong et al. 2011] algorithm is a state-of-the-art algorithm to construct an approximate k-NN graph, i.e., a graph connecting each vector of the collection to its k closest neighbors within the same collection. This remarkably simple approach is extremely effective for constructing high-quality graphs in sub-quadratic time. In this context, it is worth mentioning that a new class of approaches has recently emerged as an alternative to direct cell-probe methods. More specifically, the Kgraph [Dong et al. 2011] and the small world graph methods [Malkov et al. 2014, Malkov and Yashunin 2016] employ a graph to index a set of entities and perform a search. This class of methods works for arbitrary metrics and is state-of-the-art on benchmarks comprising millions of vectors, as shown by the top performance achieved by the Nmslib library⁷ on benchmarks

7. <http://github.com/searchivarius/nmslib>

of this size. The limiting factor of this approach is the high computational time needed to construct the index, and its large memory requirement, which currently limits its usage to medium-sized datasets.

5.7 Conclusion

The field of similarity search has evolved rapidly in the last years. On this line of research, many recent works have been guided by a compression point of view. By evidencing and exploiting the close relationship between indexing and compression problems, this point of view has become prevalent in the recent literature and is replacing sketch-based methods based on binary codes. However, it is worth mentioning that the two points of view are not necessarily concurrent, as shown by Douze et al. [2016], who propose some codes that can be employed either as binary codes and compared with the Hamming distance, or as quantized codes with compressed-domain distance estimation.


The combination of the cell-probe model with codes is currently the most effective strategy for very large collections when memory becomes an important constraint [Li et al. 2016b]. It is now adopted by many groups and the best approaches are derived from it for a very large-scale scenario. According to a talk by Google at ICML, a similar architecture was employed in Google's Goggles search engine.⁸ However, the recent graph-based approaches have settled the new state of the art for medium-size collections (i.e., for millions of vectors) when the training time and memory are secondary criteria [Li et al. 2016b]. Additionally, they offer the advantage of being effective for arbitrary metrics without the need for an embedding function. Scaling these methods to larger collections is an open problem.

8. http://en.wikipedia.org/wiki/Hartmut_Neven and <http://techtalks.tv/talks/54457/>



PART

**HUMAN-CENTERED
MULTIMEDIA
COMPUTING**



Social-Sensed Multimedia Computing

Peng Cui (Tsinghua University)

Multimedia computing technology, as one of the most effective and pervasive technologies in modern society, plays irreplaceable roles in bridging user needs with vast amounts of multimedia information. It is hard to imagine life and work without today's multimedia platforms. It is the fast advancement of multimedia computing technology that supports the development of the Internet and various web applications, which not only significantly broadens the scope of information and real life resources [Jain and Sonnen 2011] that people can reach and boost the efficiency of information seeking, but also revolutionizes society in every area from commerce to healthcare to education. In an era filled with buzzwords like big data, mobile Internet, and artificial intelligence, where will multimedia computing technology go next?

Throughout the historical development of computing technology, we can clearly observe two stages. The first is the *data-centric computing* stage, where calculation, storage, and transmission were the research foci. After the industrial revolution, people started to seek new automatic technologies that could further assist human efforts. Some of the demands for technologies involved solving the following problems: how to do fast calculation, how to efficiently store the relatively large-scale data into the very expensive digital storage medium of that time, and how to transmit the data from one place to other places on demand. With such requirements, many computing technologies such as data codec, data transmission, and the infrastructure of the Internet were born. One common characteristic of these

technologies was that most of them were established based on the Shannon Information Theory, where semantic-agnostic is a basic assumption. They focused on data itself but cared less about the content and semantics that the data represented. Thus we refer to them as data-centric computing technologies. With the rapid growth of the Internet and digital devices, the volume of data reached such a level that people had to find a way to effectively and efficiently manage and utilize them. In order to realize this, it became necessary to move up from the data level to the content level to address this content understanding problem. Then several new lines of technology such as database, information retrieval, computer vision, and multimedia analysis rapidly emerged, which we call *content-centric computing* technologies. The search engine is one of the most notable applications that was born in that period.

In recent years, the landscape of the Internet has changed significantly. In contrast to the traditional information-dominated web, people have become a new and unignorable dimension of the Internet. The emergence and popularity of UGC (user-generated content) sharing platforms, online social networks, mobile Internet, wearable devices, etc. are strong evidences for the view that the current web has become human-centric, where most information is *about people* (e.g., user profiles, social networks, and self-expressive UGC contents), generated and distributed *by people*, and ultimately exploited and utilized *for people*. The fact that referral traffic from Facebook surpasses Google for most websites¹ also demonstrates that the human-centric social network platforms have become the primary channel for information seeking. This substantial change raises a fundamental problem for multimedia computing technology: how to understand people, predict people's intentions, and proactively connect multimedia resources with people according to their real needs. With this goal, it is necessary to shift our research focus from content-centric computing to *human-centric computing*. (Figure 6.1 shows this development process and trend of computing technology.)

Unlike our understanding of data and information, today we still do not have adequate scientific approaches to computationally model people in different levels ranging from individuals and social groups to the population level. With the web evolving into the human-centric phase, data about people are deeply and comprehensively recorded at an unprecedented level that enables us to conduct systematic and thorough studies about understanding people in online platforms. Among various interesting problems, such as user profiling, behavioral modeling, and so-

1. <http://www.business2community.com/brandviews/mainstreethost/facebook-vs-google-battle-referral-traffic-01470751#AoIVW30ijKqIhgsD.97>

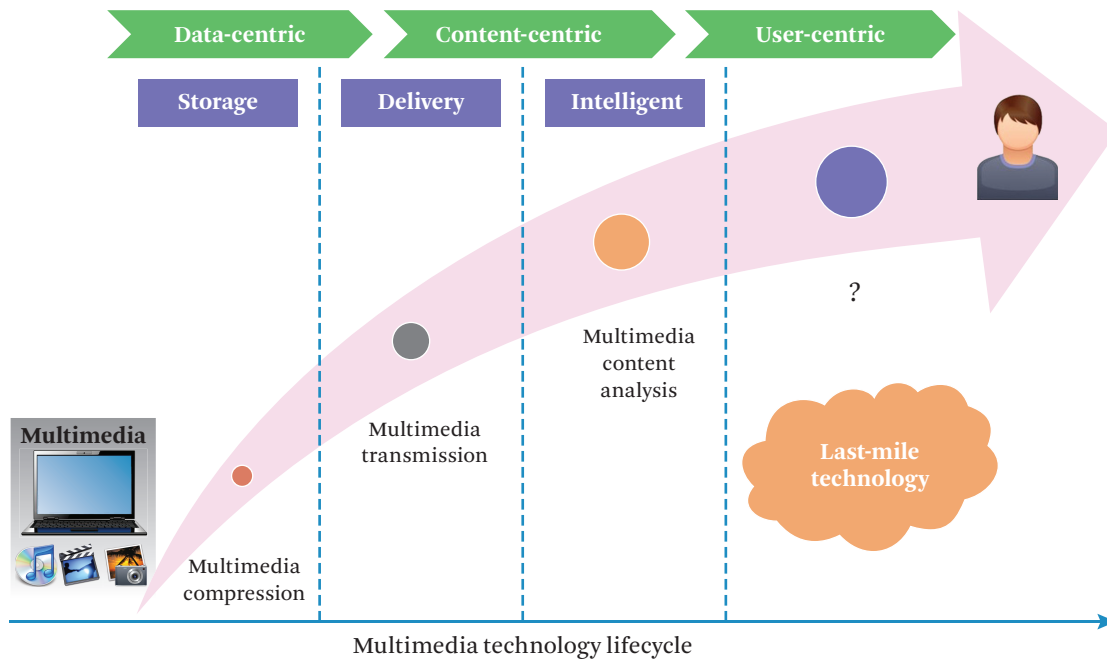


Figure 6.1 The development process and trend of multimedia technology.

cial group analysis, user intention inference plays the vital role of bridging user need with appropriate multimedia resources. If multimedia codec and transmission technology enables delivery of multimedia data to users, and content analysis technology makes it possible to extract semantics from multimedia data, then we can regard the user intention inference technology as the last-mile technology in the multimedia computing lifecycle, as it tells what multimedia resources users need, and thus closes the loop of delivering these resources to end users according to this information need.

Before jumping into the development of solutions, let us first discuss the problem of intention and its difference from traditional semantic problems.

6.1 Semantic Gap vs. Need Gap

Semantics is the study of meaning. The semantic gap² (see Figure 6.2) is often referred to as the gap between the formal low-level representations in computational

2. http://en.wikipedia.org/wiki/Semantic_gap

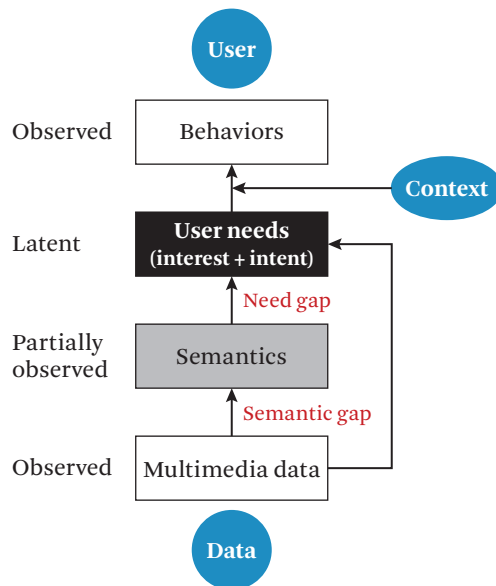


Figure 6.2 Semantic gap and need gap.

machines and the richness of high-level semantic meanings in the human mind. Bridging the semantic gap, especially in unstructured multimedia data such as visual and acoustic information, is the ultimate goal of content-centric computing technologies. Quite a number of research communities have devoted much effort to understanding *what the data represents*—e.g., recognizing objects in visual images, identifying targeting events in video sequences, and measuring textual semantic similarities in different levels from words to documents. Although there is still a long distance from our current status to ideally bridging this gap, the success of search engines and many vertical applications such as face detection and recognition have demonstrated the rapid advancement in this direction.

As mentioned before, understanding user needs is critical for achieving the ultimate goal of multimedia computing. In the scope of information science, user needs can be understood as a user's desire to obtain information to satisfy his/her conscious needs (e.g., when a user has a question in mind and goes to a search engine for answers [Hanjalic et al. 2012][Kofler et al. 2014][Teevan et al. 2008]) or unconscious needs (e.g., when a user surfs a video-sharing platform for entertainment where he has no explicit information needs in mind but just explore for interesting content [Cui et al. 2014b]), and can be further specified into interests

and intents where interest represents long-term user needs and intent represents instantaneous user needs. User needs are often latent and cannot be directly observed. How to infer or even predict the latent user needs from observed data is challenging. A basic hypothesis is that user needs will be triggered in certain situations and then will be manifested as behaviors. Thus behaviors can be regarded as the reflection of user needs. Regarding the user–multimedia interaction behaviors, on one side, they depend on the semantics of the multimedia data, as users have different preferences on the meanings of the multimedia content in nature; on the other side, the mapping from semantics to user needs is complicated and the learning of this mapping is ill-defined. Thus, between the semantics of multimedia data and user needs over multimedia data, there also exists an obvious gap: the need gap (see Figure 6.2).

Both semantic gap and need gap are critical in multimedia computing. However, semantic gap attracts much more research interest than need gap in the multimedia community. A question: can user needs be straightforwardly derived from semantics? The answer is no. It is well accepted that user needs can be represented by a distribution over semantics, but the distribution is heavily dependent on the context. For example, which videos a user wants to watch vary greatly during working time vs. leisure time, in different moods or physical conditions. Considering the implicitness of user needs as well as the incompleteness and uncertainty of observed user behaviors, how to discover the mapping mechanism between multimedia data and user needs, and its coupling with the rich context information, requires more deep and comprehensive research. Another argument might be: if the multimedia data can be replaced by semantics, which are often textual words or sentences, the need gap problem will become irrelevant. However, we argue that the multimedia data cannot be fully abstracted by semantics. Visual styles, visual impacts, and psychovisual factors cannot be accurately and comprehensively represented by semantics, but they play important roles in deciding the users' interests in multimedia data, especially considering the fact that most users watch multimedia content for entertainment or exploratory goals. For example, a user searching images of cats can seldom clearly describe why he or she prefers one image of a cat to another one purely by semantics. Intuitively, the feeling from visual factors often dominates this kind of decision. In this sense, both semantics and multimedia contents should be jointly considered in bridging the need gap. The need gap problem can be formulated as seeking a mapping function between multimedia content (represented by both semantics and visual factors) and the observed user behaviors.

6.2 Social-Sensed Multimedia Computing

The ultimate goal of multimedia computing is to deliver multimedia content to users according to their information needs (intentions). Multimedia computing can be decomposed into various stages: multimedia compression (for storage), multimedia communication (for delivery), and multimedia content analysis (for intelligence), as shown in Figure 6.1. Among these, multimedia compression and communication are comparatively well established. Since the end of the last century, multimedia analysis has become mainstream in the multimedia community, and related technologies have advanced significantly. However, how to bridge the multimedia content with end users, the *last-mile* technology for multimedia services, is rarely researched. This negligence directly causes an obvious *need gap* between multimedia data and the real information needs of users, which has become a bottleneck in advancing intelligent multimedia computing technologies for use in real applications.

At the ACM International Conference on Multimedia 2012, the twentieth anniversary of ACM Multimedia, Klara Nahrstedt and Malcolm Slaney co-organized a panel called “Coulda, Woulda, Shoulda: 20 Years of Multimedia Opportunities,” and invited several pioneering and leading researchers in the multimedia community, including Dick Bulterman, Ramesh Jain, Larry Rowe, and Ralf Steinmetz, as panelists. Among the various and interesting topics discussed, one of the most thought-provoking and sobering questions was the following: “Multimedia analysis has been a hot topic in the multimedia community for dozens of years, but why have new multimedia systems and platforms that have exploded in popularity (such as Flickr, YouTube, Instagram, etc.) not been founded by people in the multimedia community, and further, why have these systems not leveraged advanced multimedia analysis technologies?” Meanwhile, there was another more incisive and contradictory panel, themed “Content Is Dead; Long Live Content!,” which once again pushed attendees to introspectively consider the research performed on multimedia content analysis in recent years. There were various, sometimes contradictory arguments during and after the conference, yet one of the well-accepted opinions was that researchers and engineers have always assumed users like some multimedia content and applications, but have rarely tried to understand their real needs concerning multimedia data. Although the relevance feedback technology incorporated users’ interactions in image retrieval, its reliance on users’ explicit feedback is somewhat against the habit of *lazy users*, which makes it challenging to be applied in practice. User intention dis-

covery and modeling needs to be treated as a first-class citizen in the multimedia community.

It is the right time to re-consider the traditional multimedia computing paradigm, which is either *data-centric* or *content-centric*. Previous research has provided us with strong capability for processing and understanding multimedia data and content. At this point, as a discipline mainly targeting the technologies and services for end users, we should also pay more attention to user needs. How to transform data-centric or content-centric multimedia computing into *user-centric multimedia computing* presents great challenges and opportunities for both academia and industry.

Data specifying user needs is absolutely necessary to understand real user needs for multimedia. An alternative to explicitly surveying user needs, which is often costly and impractical, is to collect multimedia data, user information, and the interaction behaviors between users and multimedia data, from which we can implicitly and continuously discover the user needs for multimedia. However, these data types were not readily available in the past, since most of the interaction behaviors between users and multimedia were unobservable. Fortunately, with the emergence of social media platforms (such as Flickr, Facebook, YouTube, etc.), billions of users proactively interact (e.g., generate, share, comment, etc.) with huge volumes of multimedia data, and these interaction behaviors are being recorded at an unprecedented level. Thus, social media has eventually formed a valuable pool of data about user needs, which provides the precious opportunity to bridge the need gap in multimedia computing.

More specifically, users' (both crowds' and individuals') intention-related information (including long-term interests, instantaneous intentions, emotions, etc.), their behavior patterns, and ultimately, the common principles of user-multimedia interactions under different contexts can all be sensed from social media, and summarized as *social knowledge on user-multimedia interactions*. It is such social knowledge that reflects user needs and establishes a bridge between multimedia data and these needs. How to organically integrate multimedia data, user needs, and social knowledge into multimedia computing technology is a critical issue. Thus, we propose a new paradigm, *social-sensed multimedia computing*, to bring social media, viewed as a valuable source of sensing user needs and social knowledge, into the loop of multimedia computing, as shown in Figure 6.3. We believe this new paradigm will naturally transform the landscape of multimedia computing from traditional *data-centric* or *content-centric multimedia computing* to *user-centric*

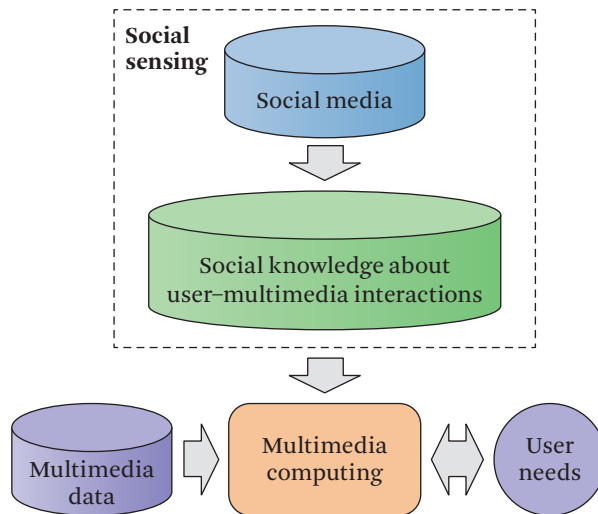


Figure 6.3 Illustration of the social-sensed multimedia computing paradigm.

multimedia computing, which will improve users' experiences in various multimedia applications and services.

6.3 Basic Problems and Key Technologies

As a new scheme, social-sensed multimedia computing faces some new basic problems. In comparison to traditional scheme, the new problems mainly relate to social multimedia representation, user modeling, user-multimedia interaction analysis, and the integration with traditional multimedia computing methods. Social media data collection has also become a non-trivial preconditioned problem.

6.3.1 Reliable Data Collection of Social Media

Different data will lead to different conclusions. Due to the huge volume of social media data, the collection of a reliable dataset is crucial in order to safely draw conclusions on multimedia intention patterns. Since most social media platforms set various limits over data APIs, it is almost impossible for researchers to obtain a complete dataset of a social media platform. This raises a series of problems. For example, how comprehensive is the collected data, and is it representative enough? In [Morstatter et al. \[2013\]](#), the data collected from the Twitter Streaming application programming interface (API) was shown to be quite different from the Firehose data of Twitter in various aspects. [De Choudhury et al. \[2010\]](#) and [Golub and](#)

Jackson [2010] indicated that selection/sampling bias can significantly influence the discovery of user behavior patterns in social media. Thus, the question of how to design and develop a new data collection strategy for social-sensed multimedia computing requires further attention.

6.3.2 Social Representation of Multimedia Data

Social multimedia was first defined by Mor Naaman as “an online source of multimedia resources that fosters an environment of significant individual participation and that promotes community curation, discussion and re-use of content” [Naaman 2012]. The main characteristic that differentiates social multimedia from traditional multimedia is that the former boasts significant user participation and interactions with multimedia content. For example, users tag images in Flickr, add comments to videos in YouTube, and Like or Dislike videos and images in Facebook. These are all important resources for discovering patterns of user behaviors toward multimedia content. However, all of the current multimedia representation methods (e.g., low-level features, concepts, and visual attributes), are designed to bridge the *semantic gap*. There remains an obvious gap between semantics and user responses and behaviors. Therefore, new representation methods for social multimedia must be found to simplify mapping between multimedia content and user responses and behaviors.

6.3.3 User Profiling and Social Graph Modeling

The concept of the user dimension is quite new for the multimedia community, yet it has been widely investigated for years in data mining and information retrieval communities, with its applications ranging from individual user modeling to social graph modeling. More specifically, user profile inference, social graph analysis, and tie strength measurement have become very popular in the social network analysis field. However, most related research has been based on text information. When we attempt to transfer that into social-sensed multimedia computing, several fundamental issues arise: Can the user and social knowledge learned from text data be adapted to multimedia data? Is multimedia data able to tell us a different (or a more complete) story about the characteristics of users and social relations? Intuitively, multimedia content has intrinsically different structures and feature spaces from text content, and can typically provide much richer semantics and meanings. To guarantee that the learned user profiles and social graph models can be seamlessly bridged with multimedia data, we must revisit user profiling and social graph modeling in the context of social multimedia environment.

6.3.4 User-Multimedia Interaction Behavior Analysis

The major goal of the sensing part of social-sensed multimedia computing is to discover user-multimedia interaction behavior patterns, from which user intentions toward multimedia data can be inferred. User-multimedia interaction behaviors should be investigated at different scales depending on the level of support required by various multimedia applications. In particular, these interaction behaviors can be categorized into *microscopic*, *mesoscopic*, and *macroscopic* levels, which respectively correspond to the interaction behaviors with multimedia data of individual users, groups of users, and global users. Microscopic analysis can be specified as, but not limited to, user interest modeling and user sentiment analysis, which can support personalized search and recommendation for multimedia. Mesoscopic analysis includes collective behavior analysis, social influence modeling, and so on, which can support social multimedia marketing, and socially aware multimedia communication. Macroscopic analysis broadly refers to multimedia propagation analysis and prediction in a social environment, which can support multimedia popularity prediction and social multimedia monitoring. These all make it possible to infer user intentions with regard to multimedia data, and thus play significant roles in social-sensed multimedia computing.

6.3.5 Heterogeneous Information Integration

We inevitably confront the problem of heterogeneous information when trying to bridge social media with traditional multimedia computing methods. Let us take image search as an example. Traditional image search systems mainly include textual query and image information, and have well-established frameworks. After introducing social media into the loop, we have to consider two important problems. First, after sensing the information and knowledge from social media, into which step in the traditional framework should we inject the query step, the re-ranking step, or other steps? Second, how can we effectively and efficiently integrate heterogeneous information such as textual queries, image content, social graphs, user profiles, and behavior patterns, to make the results returned more consistent with user intentions? The answers to these two problems depend heavily on the specific multimedia application, yet overall the heterogeneous information integration strategy should be subtly designed to balance the virtues of social-sensed multimedia computing with increased computational and storage costs.

6.4 Recent Advances

Social-sensed multimedia computing is a quite new research direction, and has attracted a number of researchers. Some of them have made valuable attempts

and significant progress on the basic problems, and others have demonstrated the value of social-sensed multimedia computing in different application scenarios. Here, we discuss some representative works that exemplify representative research in line with social-sensed multimedia computing.

6.4.1 Demographic Information Inference from User-Generated Content

How to align multimedia data with social aspects of users is essential for social-sensed multimedia computing. Among various intermediate factors, demographic information in particular has received significant research interest in recent years. If we can establish the mapping relationships between multimedia content and user (i.e., viewer) demographic information, then we can predict user demographic information based on their multimedia consumption history, understand multimedia content assisted by the aggregated demographic information of a viewer, and bridge multimedia content and users by demographic information. On this topic, [Ulges et al. \[2012\]](#) received the best paper award of the International Conference on Multimedia Retrieval (ICMR) 2012. They employed user comments and user profiles on YouTube and revealed a strong correlation between viewer demographic information and semantic concepts appearing in viewed videos. In another study, [Guo et al. \[2013\]](#) proposed an approach to infer whether the author is amateur or professional from the contents of uploaded videos. [Zhong et al. \[2013\]](#) proposed a multi-task learning method to predict user demographics based on mobile data, and reported quite high prediction accuracies for different demographics.

These works demonstrated the viability of linking multimedia content with user demographic information and paved the way for bridging users and multimedia data. However, demographic information is after all only a part of user profiles. How to represent user profiles in a richer way with finer granularity to better interpret user interaction behaviors with multimedia data is still a worthy topic for further research.

6.4.2 User Interest Modeling from Multimedia Data

From a user perspective, interest is a major endogenic factor that drives interaction behaviors with information. A significant body of research has focused on user interest modeling based on user search logs and text information. [Qiu and Cho \[2006\]](#) represented user interest by topics and proposed a method to learn user preferences from past query click history in Google. [Agichtein et al. \[2006\]](#) proposed a method to learn the user interaction model with which user preference over the search results can be predicted. [Teevan et al. \[2005\]](#) explored rich models for interest modeling by combining multiple resources, such as search-related information, user-relevant documents, and emails. More recently, [Jiang et al. \[2012a\]](#) and [Cui](#)

et al. [2011] investigated user-information interaction behavior patterns in social network environments.

The interest modeling problem is more challenging in the image domain due to the high-dimensional space and the semantic-gap problem. Lipczak et al. [2013] analyzed user favorite behavior patterns in Flickr. Xie et al. [2005] proposed detecting user interests from user-image interaction behaviors recorded by image browsing logs. Yang et al. [2013] investigated the emotion prediction problem for individual users when watching social images. Tags of images are mined to construct the topics and ontology to represent user preferences [Negoescu and Gatica-Perez 2010]. Similar to the problem that user intentions cannot be well represented by query words in image search [André et al. 2009], user interests in images cannot be well represented by tags, either. Visual factors, such as visual style and visual quality, eventually play important roles in user interest formation. The majority of the user interest-modeling methods developed thus far have relied on text information. How to discover user interest patterns related to multimedia content is still an open, unresolved issue.

6.4.3 Intention-oriented Image Representation Learning

Learning image representation by deep model has recently made remarkable achievements for semantic-oriented applications, such as image classification. However, for user-centric tasks, such as image search and recommendation, simply employing the representation learnt from semantic-oriented tasks may fail to capture user intentions. Aiming at capturing user intention to improve the performance of image retrieval and recommendation, query log analysis [Jansen 2006, Hua et al. 2013, Pan et al. 2014] and relevance feedback [Rui et al. 1998, Zhang et al. 2012] have been proposed in the past years. However, users' query log in image search engines can hardly be accessed for common users and researchers. Besides, the frequent operation in relevance feedback methods may sometimes reduce user satisfaction. With the development of social media, the information in social platforms, such as image tags [Li et al. 2009], user behaviors [Yuan et al. 2013], and user relationships, are utilized to analyze user interests. Cui et al. [2014a] proposed a social-sensed image search framework, which first summarizes user interests based on favorite images in Flickr, and then reranks the search results based on interests to realize personalized search. Liu et al. [2014] learned an image distance metric based on social behavioral information to evaluate image similarity of user intention. However, traditional works based on social information usually use "shallow models." Thus, their ability to bridge semantic gap and need gap can still be strengthened by deep models. Although Yuan et al. [2013] explored learn-

ing latent features of social entities (e.g., users, images, tags) by deep model, this work only focused on learning the relationship between pair-wised social entities (e.g., user-image and image-tag). For an image that has multi-modal information (such as faves and tags), there is no common representation for this image. In other words, when user intention is learnt based on user-image relationship, the semantic meaning in the image-tag relationship will be ignored. Furthermore, as other social-sensed works show [Liu et al. 2013a], the pair-wise relationships in social platforms are usually unreliable because social information is very sparse and noisy. To make image representation robust, it must completely capture both semantics and user intention. In Liu et al. [2015], a novel Socially Embedded Visual Representation Learning (SEVIR) approach is proposed, where an asymmetric multi-task CNN (amtCNN) model would embed a user intention learning task into a semantic learning task. With the partially shared network architecture, the learnt representation could capture both semantics and user intentions.

6.4.4 Intention-oriented Distance Metric Learning

Image distance (similarity) is a fundamental and important problem in image processing. However, traditional visual features-based image distance metrics usually fail to capture human cognition. Traditional metric learning research usually aims at learning metrics from labeled examples. The methods can be categorized into supervised ones [Yang and Jin 2006] and semi-supervised ones [Hoi et al. 2008]. In supervised metric learning, labels of images are complete, such as the categories of the images. Weinberger and Saul [2009] proposed a method named large margin nearest neighbor (LMNN), which aims at reducing the margin of nearest neighbors. In semi-supervised metric learning, we do not have all the labels but only know some pairs of images are similar and some pairs are dissimilar. Thus, these methods aim at reducing the distance among the similar set and enlarging the distance among the dissimilar set. In our work, we do not have any labeled images but the images with social behavioral information. Although the social similarity can be evaluated by the social information, its reliability is not guaranteed because the social data are very noisy and uncertain. In addition, social similarity is a wholly new dimension to evaluate image similarity and it is very sparse. Thus visual distance needs to be maintained when an image does not have a socially similar neighbor. In Liu et al. [2014] a novel Social-embedding Image Distance Learning (SIDL) approach is presented to embed the similarity of collective social and behavioral information into visual space. The social similarity is estimated according to multiple social factors. Then a metric learning method is especially designed to learn

the distance of visual features from the estimated social similarity. In this manner, we can evaluate the cognitive image distance based on the visual content of images.

6.4.5 Multimedia Sentiment Analysis

Sentiments (or emotions) reflect user attitudes toward information, and capture aspects of users different from those revealed by user interest data. For example, a user might be interested in a set of social news, but may express different emotions toward different news items. Sentiment analysis can therefore help to obtain an insightful understanding of user behavior patterns related to multimedia data. Although recent progress has been made in text-based sentiment analysis, efforts for multimedia-based sentiment analysis lag far behind. The most relevant research in this direction has incorporated analysis of aesthetics and emotions in images, where most work has attempted to predict sentiment from low-level visual features [Joshi et al. 2011]. In addition, Borth et al. [2013] recently proposed and published a large-scale visual sentiment ontology and a visual concept detector library to support the detection of sentiments from images.

The common goal of these studies was to detect and predict general and objective emotions from images. However, sentiment is a highly personalized and subjective issue. The sentiment expressed by an image varies across different contexts, cultures, and even viewers. This point is quite critical for social-sensed multimedia computing, as social media can provide the opportunity to incorporate social, contextual, and personal differences into sentiment analysis. Yang et al. [2013] made an initial attempt to integrate social and personal factors for emotion prediction, and demonstrated that combining social-personal factors and visual-textual features was highly important. Their work deserves further exploration; advances in the field will help us discover intrinsic mechanisms of sentiment formation of users when exposed to multimedia content, and accurately predict sentiments for different users under different contexts in different cultural environments.

6.5 Exemplary Applications

In preceding sections we have described the basic problems in social-sensed multimedia computing, and reviewed some representative recent advances and solutions in the field. In this section, we focus on how solutions to social-sensed multimedia computing can have real-world impacts. More specifically, we describe how such impacts can improve the performance of traditional multimedia computing technologies, and more broadly, play an important role in improving user experiences in multimedia services.

6.5.1 Social-sensed Multimedia Search

According to the survey in Smyth [2007], approximately 50% of search sessions still fail to find satisfactory results. The lack of understanding of user intent is one of the key causes. In recent years, the discovery of user intent in multimedia search has attracted significant research interest. However, the majority of studies have focused on constructing user models from social media and applying them in vertical searching in social media. How to generalize user models sensed from social media to assist general multimedia search remains an untouched problem. Chang [2013] stated that in today's highly connected world, knowing what is being written, favored, or shared would enrich the determination of how content could be indexed and searched. This statement resonates well with the idea of underpinning social-sensed image searches, where user profiles and behaviors in social platforms are sensed, harnessed, and shared to adapt the results of general multimedia search engines. Merging search engine and social media has clearly become a common trend in industry: for example, Google has acquired YouTube and launched Google Plus, Yahoo has acquired Flickr, and Facebook has put forth efforts to develop search services with a Facebook-external scope. The following is a reasonable novel multimedia search scenario to consider: a user conducts an image search in Google by inputting the query together with his/her Google Plus ID. The Google search engine can then derive the user's personal data from Google Plus, analyze the user's interaction behaviors with multimedia data, evaluate the user's intent, and re-rank the image search results in a personalized way. Much could be leveraged by integrating social media platforms with multimedia search systems. How to discover and represent user search intention from social media and seamlessly bridge these user intentions with multimedia search systems is a research issue in need of serious attention.

6.5.2 Social-sensed Multimedia Recommendation

Recommender systems are becoming increasingly important because of the overload of information brought about by today's Internet. Video recommender systems are required, in particular, because of the high time costs of watching videos. Netflix reported that 75% of the content that people watch follows a recommendation. These video recommender systems therefore play the role of a bridge between users and videos. In the literature, collaborative filtering (CF) has achieved great success in recommender systems. User-based CF methods represent users with videos as features, such that user-video matching can be conducted in the item space. In contrast, item-based CF methods represent videos with users as features

and calculate the matching degree of user–video pairs in user space. However, the performances of these methods are seriously affected by the sparsity of the user–video collaborative matrix; they are unable to infer meaningful information about videos (or users) that lack interactions with different users (or videos). More recently, matrix factorization based CF has become more popular. It assumes a common low-dimensional latent factor representation for both users and items such that the user–item matching degree is measurable in the latent space. However, the latent factors are hardly interpretable, which makes it difficult to generalize the learned representations to new data. In addition, all CF methods suffer from the cold start problem, making recommendations for new users or new videos difficult, owing to a lack of information in the collaborative matrix.

Fortunately, the emergence of social media brought us a vast amount of users, videos, and the observable interaction behaviors between users and videos. Recognizing that influence is a subtle force that governs the dynamics of social networks, influence-based recommendation [Leskovec et al. 2006] involves interpersonal influence in social recommendation cases. Trust-based approaches [Jamali and Ester 2009] exploit the trust network among users and make recommendations based on the ratings of users who are directly or indirectly trusted. Jiang et al. [2012a, 2014a] proposed a probabilistic factor analysis framework, which fuses users' preference and social influence together. Furthermore, Jiang et al. [2012b] investigated the social recommendation problem in a multiple domain setting. There is still a vast space to explore in exploiting social information to boost the multimedia recommendation performances in both accuracy and interpretability.

6.5.3 Social-sensed Multimedia Summarization

Due to the explosive growth of multimedia data, it is quite difficult for current multimedia summarization technologies to conduct their information assimilation processes simply. Traditionally, the goal of multimedia summarization is to select as few representative frames (for videos) or images (for image sets) as possible to give users the main, summarized content of these multimedia items. The challenge is that different users may be interested in different parts of the same multimedia item. Therefore, a user's motivation for watching multimedia content is not limited to fast browsing of the complete multimedia item; it also includes the intention to enjoy semantically meaningful content that reflects both important video content and a user's interests. Users often skip multimedia items if their summaries do not present sufficiently interesting content. How to discover engaging content for different users and to generate different summaries according to their interests is of paramount importance for improving the diffusion of multimedia items and the user experience in consuming videos and images. How to

implicitly discover user interests regarding visual contents and adapt multimedia summaries accordingly is still an open research problem. Traditionally, the interactions of users and multimedia items are not observable, which makes it difficult to discover user interests. The emergence of social media in recent years, in which users actively share their personal information and their interactions with multimedia items, means their preferences are readily available. This trend can help actualize the idea that underpins social-sensed multimedia summarization, where user interests are discovered from their profiles and interactions with multimedia items in social media platforms, and interest models thus created could be used to adapt multimedia summarization results. Such applications will involve heterogeneous media and user information, including user profiles, user behaviors, social graphs, and most importantly, visual content. How to discover user interests from vast amounts of heterogeneous information and effectively match user interests with multimedia content is the fundamental issue standing in the way of such applications. Furthermore, suitable computational efficiency and scalability of this kind of algorithm or system is another critical issue.

6.5.4 Social-sensed Video Communication

Online social networks have reshaped the way video content is generated, distributed, and consumed on the Internet. Given the massive number of videos generated and shared in online social networks, it is convenient for users to directly access video content in their preferred social media platforms. An intriguing exercise would be to study how service is provided for social video content to global users who report satisfactory quality-of-experience. Propelled by this idea, [Wang et al. \[2012b\]](#), the winner of the Best Paper Award of ACM Multimedia 2012, proposed a propagation-based socially aware video replication strategy. They sensed and summarized the characteristics of video propagation patterns in social networks, including social locality, geographical locality, and temporal locality. Motivated by these insights, they proposed a propagation-based socially aware replication framework using a hybrid edge-cloud and peer-assisted architecture, and demonstrated that the local download ratio in the edge-cloud replication and the local cache hit ratio obtained through peer-assisted replication could be improved by 30% and 40%, respectively, by exploiting the propagation patterns sensed from social media.

6.6 Discussions on Future Directions

The user dimension is new but critical for the multimedia computing community. To bridge the need gap, we need to invest more efforts in understanding the users, both individually and collectively. In this chapter, we have proposed the

social-sensed multimedia computing paradigm, and advocated the necessity to organically integrate social network and social media data with multimedia computing tasks. We have summarized the basic problems in this direction, reviewed some representative works, and introduced the exemplary application scenarios of social-sensed multimedia computing. Overall, we see the trend that more and more researchers in multimedia community are focusing on the user dimension and making fast advancement in this line of research.

In the future, there are several important research issues that need to be addressed.

6.6.1 Social Attributes for Users and Multimedia

Most multimedia analysis and vision problems are grounded in data representation and machine learning methods. In recent years, machine learning methods have received intensive research, and made much progress. However, for most multimedia applications, problems of representing multimedia data constitute a bottleneck that obstructs performance improvement. Here we raise a question: for what purpose is the multimedia data represented? In the past, we have represented multimedia data to address the semantic gap problem. The emergence of deep learning provides a promising way to solve data representation for the semantic gap, wherein images and videos can be represented by pixels. Now, the question becomes: if we hope to address the need gap problem, how should we represent multimedia data now?

Data representation is a middle layer between low-level raw data and high-level objective data. For example, when we conduct feature engineering for the semantic gap, we extract features such as colors, textures, etc., which can be extracted from raw pixel data and can somewhat be linked with high-level semantics. If we replace the objective data with user intentions, the middle-level data representation would be more challenging due to the higher abstraction required of user intentions compared with that of semantics. Fortunately, social media provides us with a vast amount of user profiles and associated observable interaction behaviors between users and multimedia items. The *Homophily* hypothesis³ suggests that extracting a middle-level representation layer to represent both multimedia data and users is possible and reasonable. In this common space, the interactions between the two can be easily interpreted. In contrast with traditional representation methods that begin development from the multimedia end, another direction to attempt is

3. Homophily is the tendency of individuals to associate and bond with similar individuals. It is often used to account for the similar behaviors of similar people toward given new ideas or innovations.

starting from the user end. Social attributes, which capture user characteristics in terms of social aspects, can be extracted from user profiles and their behaviors, and these social attributes can be further propagated to multimedia items through the interaction behaviors between users and multimedia items. The social attribute space can then be used as the common space between multimedia data and users, from which user intentions and their interaction behaviors with multimedia items can be inferred. It is worth investigating how to define, represent, and extract social attributes for both multimedia data and users, how to propagate these social attributes between the user layer and the multimedia data layer to make them aligned, and how to interpret and infer interaction behaviors between users and multimedia items in the social attribute space.

6.6.2 From Social Multimedia, Beyond Social Multimedia

The explosive growth of social multimedia content on the Internet is revolutionizing the landscape of various multimedia applications. It has even led to a new research area, called social multimedia computing [Tian et al. 2010]. The new types of multimedia content, meta-data, context information, and interaction behaviors in social multimedia present a significant opportunity to advance and augment multimedia and content analysis techniques. The majority of previous research works have regarded social multimedia as a new research objective and proposed new methods to either exploit the new types of data or solve new problems within social multimedia. Here, we argue that the significance of social multimedia with respect to the multimedia field goes far beyond the emergence of new types of data and problems. The rise of social networks and social media platforms was responsible for bringing people onto the Internet for the first time in many cases, and observable user profiles and behaviors provide us with valuable resources to discover *common principles* of interactions between users and multimedia data. The research principles here include how and why users generate, share, and assimilate multimedia data, as outlined by Chang [2013] and Hanjalic [2013]. Notably, these *common principles* should not be limited to social multimedia platforms. Instead, they and the knowledge sensed from social media should be generalizable to other multimedia applications where user profiles, behaviors, and social relations are not observable. For example, can knowledge sensed from Flickr be used to improve Google image search? Can knowledge sensed from YouTube help to design a recommendation system for TV programs? Can knowledge sensed from Flickr and Instagram be integrated to form a more comprehensive understanding of users? In summary, how to sense *transferable* and *interoperable* common principles and knowledge from social multimedia and seamlessly integrate this knowledge with

various multimedia services should be the ultimate goal of social-sensed multimedia computing. These objectives pose great challenges to those who would seek to improve current techniques in the multimedia community, and open a broad assortment of new research topics that are worth investigating.

6.6.3 From Social-sensed Multimedia Computing to Multimedia-sensed Social Computing

In the keynote talk given by Ramesh Jain in the social media workshop of ACM Multimedia 2009, he raised a thought-provoking question: It is obvious that *social applications* will significantly advance multimedia technology, but what can multimedia do *for* social applications? Today, how multimedia techniques can be improved by integrating social information is obvious, yet how to exploit multimedia information to improve social computing is still unclear. However, many social phenomena are ultimately relevant to or even driven by multimedia. For example, the music video for “Gangnam Style” owes its overwhelming popularity to macro-level social atmosphere and cultural trends that facilitate such viral bursts of distribution. As an example, how “Gangnam Style” was able to become popular and what this says about society and global culture are representative topics for social computing to address; but the analysis and characterization of this music video itself is the aegis of multimedia computing, and also important. For these purposes, the semantics expressed by multimedia data are not enough, and more abstractive concepts, such as the emotions they deliver and the trends embodied by certain multimedia elements, are inevitably required. We have witnessed an increasing volume of research in this line, including multimedia affective computing [Joshi et al. 2011], social multimedia as sensors [Jin et al. 2010], trending multimedia elements discovery [Xie et al. 2011], and others. How to make social-oriented multimedia computing techniques sufficiently reliable and integrate them into social computing methodology is still a challenging task that is critical for progress in multimedia-sensed social computing. Nevertheless, advances in these dimensions would definitely help to promote the importance of multimedia technology with respect to other disciplines, and improve the user experience and social impact of multimedia services.

6.6.4 MPEG-X: A New Standard for Human-centric Multimedia Services

As stated above, social-sensed multimedia computing, a new paradigm for multimedia computing, will significantly expand the research scope of traditional multimedia computing. Based on recent research trends, we believe that fruitful achievements will occur in this direction in the near future. How can we then distribute these techniques among various multimedia applications and platforms? Stan-

standardization, which has proved effective in the development process of video services, comes to mind. MPEG (Motion Picture Experts Group), the leading organization for standards for audio and video compression and transmission, has successfully published a series of standards, all of which are either data-centric for data compression, transmission, and forensics (such as MPEG-1–4, and the recently published standards), or content-centric for video content description (such as MPEG-7 and MPEG-21). Although these standards form the construction of good infrastructure for basic video services, none can support simple guided access to multimedia content according to personalized preferences of users. Tseng et al. [2004] proposed a framework for personalizing video using MPEG-7 and MPEG-21, but it is limited by the hypothesis that objects or concepts appearing in videos can represent user preferences, which is usually not the case. On the Internet today, users enthusiastically put their personalities on display and seek to acquire personalized information and experiences. A human-centric standard for multimedia content to meet these needs is thus highly desirable. In order to simplify matching between multimedia content and user intentions, proper representation and abstraction of the multimedia content that can be linked with user intentions is a vital component. This is the goal of social-sensed multimedia computing. The human-centric multimedia standard MPEG-X, together with novel social-sensed multimedia computing techniques, would comprise essential infrastructure for the modern era of socialized and personalized multimedia services.

6.7 Conclusion

The rise of social networks and social media platforms brought many people to the Internet for the first time, and observable user profiles and behaviors provide us with valuable resources to discover common principles of interactions between users and multimedia data. The research principles here include how and why users generate, share, and assimilate multimedia data. Notably, these common principles should not be limited to social multimedia platforms. Instead, they—and the knowledge sensed from social media—should be generalizable to other multimedia applications in which user profiles, behaviors, and social relations are not observable.

The ultimate goal of social-sensed multimedia computing should be to sense transferable and interoperable common principles and knowledge from social multimedia and seamlessly integrate this knowledge with various multimedia services. This objective poses great challenges to those who seek to improve current techniques in the multimedia community and opens up a broad assortment of new research topics that are worth investigating.



Situation Recognition Using Multimodal Data

Vivek Singh (Rutgers University)

Concept recognition from multimodal data streams is a fundamental research challenge in multimedia computing. Situation recognition is a specific type of concept recognition problem aimed at deriving actionable insights from heterogeneous, real-time, big multimedia data to benefit human lives and resources in different applications.

Media (image/audio/video) processing research has made major progress on the tasks of object recognition, scene recognition, event recognition, and trajectory recognition in the last century. Each of these concept recognition problems focuses on analyzing observable media to recognize an application-centric concept (e.g., “chair,” “office,” “intrusion”). Dealing with the specifics of the media requires sophisticated techniques for pixel analysis, noise reduction, signal processing, time-frequency transformations, and machine learning techniques. The complexities of dealing with each of these areas has led to multiple research efforts focusing on analysis of a particular medium (e.g., images).

Hence, to date most research efforts have focused on the problems of object, scene, event, and activity recognition. For example, one of the first well-known approaches for visual recognition was proposed by Elias, Roberts, and colleagues in the 1960s [[Elias et al. 1963](#)], and one of the first efforts on event recognition was made by Haynes and Jain in the 1980s [[Haynes and Jain 1986](#)]. While transformative and timely in their own rights, these efforts focused on intra-media concepts (i.e., those which manifest themselves, and can be recognized within a single media object, for example, a tree, or a chair in an image). Given the widespread availability of data and the newer, more complex environment that we live in, it is now important to define and recognize evolving concepts (i.e., those which occur in the real world,

are constantly evolving, and inherently manifest themselves over heterogeneous multimedia streams from numerous sources).

Specifically, we do not need to undertake analysis based on data coming from a single media element, modality, time-frame, or location of media capture. Real-world phenomena are now being observed by multiple media streams, each complementing the other in terms of data characteristics, observed features, perspectives, and vantage points. Each of these multimedia streams can now be assumed to be available in real-time, and an increasingly large portion of these come inscribed with space and time semantics. The number of such media elements available (e.g., tweets, Flickr posts, sensor updates) is already in the order of trillions, and computing resources required for analyzing them are becoming increasingly available. These trends are likely to continue, and one of the biggest challenges in multimedia computing in the near term is likely to be that of concept recognition from such multimodal data.

As shown in Figure 7.1, situation recognition builds on and extends object recognition, scene recognition, activity and event recognition, and complex event processing. The challenges in situation recognition are very different from those in object or event recognition. For example, we can no longer just *accept* heterogeneity, or allow multiple data streams; we need to *expect* these and capitalize on them. We need to focus on recognition of real-world phenomena based on their footprints across multiple heterogeneous media. This allows for solving practical human problems by correlating data ranging from social media to sensor networks and satellite data.

To do this, one may build upon techniques rigorously developed by computer vision and multimedia researchers, which aggregated pixel values for identifying low to mid-level features (moments, color, shape, texture, area, edges) to recognize higher-level concepts (boy, face, rose) in applications. However, the computer vision research field has mostly focused on rigorous feature extraction for classification problems; its recognition models (e.g., “chair” detector) tend to be opaque and un-editable at run time. The database field, on the other hand, has focused on transparency of information; information about any entity or concept can be obtained flexibly at run-time by declaratively defining a “model” based on attributes of the data. Hence, there is a need to combine the strength of these two directions—rigorous feature extraction and run-time model definition—to create sophisticated situation recognition systems.

Examples of relevant situations to recognize include weather patterns (beautiful days, droughts), natural disasters (hurricanes, wildfires, great monsoons), economic phenomena (booms, busts, recessions), traffic (jams, accidents, smooth

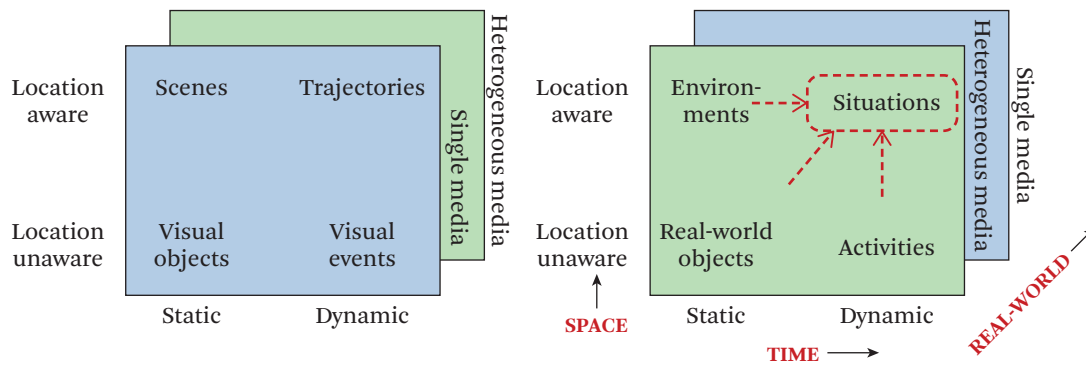


Figure 7.1 Different types of concepts can be recognized in different data availability settings. Using a single medium, such as images, can result in concepts that are rooted more in the medium than the real world, but using different media, it is possible to express concepts as they occur in the real world. (Adapted from Singh et al. [2012])

progress), seasons (early spring, winter, late fall), social phenomena (demonstrations, celebrations, uprisings, flash mobs, flocking, happiness index), and so on. The situational information derived can be used to provide information, answer queries, and also take control actions. Providing tools to make this process easy and accessible to the masses will impact multiple human activities including traffic control, health care, business analysis, political campaign management, cyber security monitoring, disaster response, crisis mitigation, and homeland security.

However, the progress in generating actionable insights from diverse data streams is still slow and the field of situation-aware computing is in its infancy. This chapter summarizes a recent line of work undertaken by the author and colleagues (e.g., [Singh et al. 2014, Gao et al. 2012, Singh et al. 2010d, Pongpaichet et al. 2013, Singh and Jain 2016]) to tackle some of the challenges in the field of multimedia situation recognition.

The organization of this chapter is as follows. Section 7.1 discusses a motivating example and the current problems in building situation-aware systems. Section 7.2 surveys possible definitions of situations across fields and provides one computational definition of situations as relevant to multimedia computing. Section 7.3 describes a framework for situation recognition that emerges out of a recent line of work by the author and colleagues. Section 7.4 describes EventShop, which is an open-source toolkit for situation recognition. Section 7.5 discusses the creation of situation-aware applications using EventShop, and Section 7.6 discusses open challenges and research directions.

7.1 The Emerging Eco-system and a Motivating Application

As shown in Figure 7.2, the Cloud today connects a variety of data streams related to multiple human functions including traffic, weather, and health. These data are in archived databases as well as in the form of real-time streams reporting attributes from different parts of the world. The real-time streams originate either from the traditional sensor/device-based sources (e.g., PlanetarySkin, satellite imagery), or the increasingly common human reporting mechanisms (e.g., Twitter and Facebook status updates). All these data can be aggregated in the Cloud and used for situation recognition and action taking.

Humans play multiple roles in this eco-system. *Human sensors* can describe different aspects of a situation, many of which are not yet measurable by any hardware sensors. Millions of users are already acting as *human actuators*, getting daily alerts, advices, and recommendations for undertaking different actions. This trend will only increase [Sheridan 2009, Dimandis 2012]. As demonstrated in the “wisdom of the crowds” [Kingsbury 1987] concept—or Wikipedia as a system—different users can act as *wisdom sources* and work toward completing different tasks including the configuration of applications for different situations [Singh et al. 2009b]. Lastly, *analysts* can visualize and analyze different situations to undertake important macro-decisions affecting their country, state, county, or corporation.

Together, this eco-system allows for the generation of unprecedented volumes and variety of data, while also allowing multiple human beings to contribute their

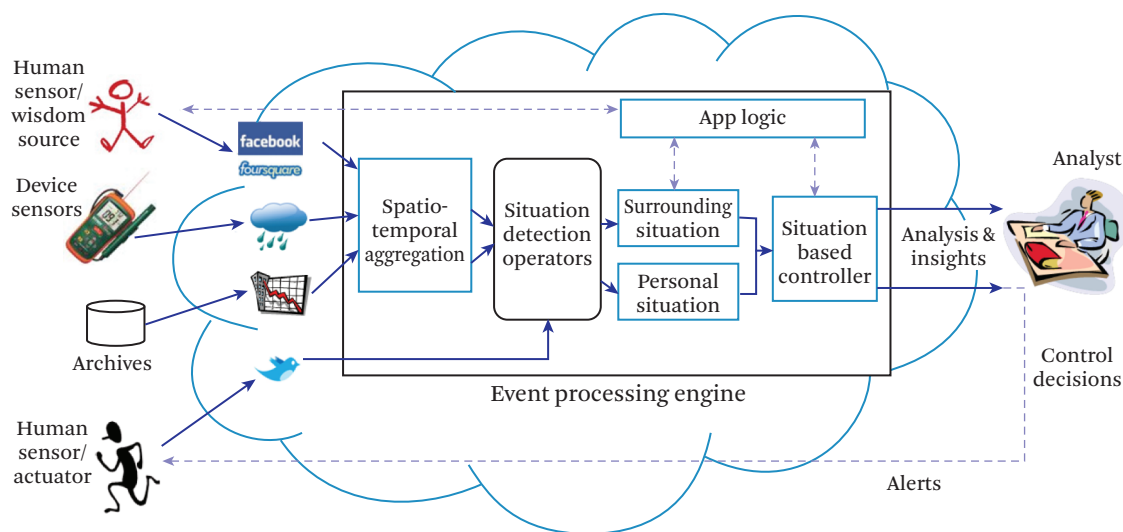


Figure 7.2 The emerging eco-system. (Adapted from Singh and Jain [2016])

wisdom for supporting various applications. Let us consider one representative application in this eco-system.

7.1.1 Motivating Application: Asthma Risk–Based Recommendations

Alice, a part-time *Ushahidi.com* volunteer and full-time mother of a child affected by asthma is worried about the seasonal asthma outbreaks. She wants to create an evolving map of risk for people with asthma across the United States. Not knowing how to define asthma-risk level, she approaches her friend Bob, who is an expert on respiratory diseases. He explains that asthma-risk depends on multiple factors, both personal and environmental. As a first step he advises her to focus on three environmental factors: sulphur dioxide concentration in the air, nearby polluting factories, and the number of active allergy cases reported. He advises her that the pollution data can be obtained from a U.S. Geological Services website, the number of active cases can be obtained from Twitter, and nearby factory data from www.epa.gov. He helps her create a conceptual blueprint of how these data values can be combined to classify the U.S. into zones of low, middle, and high asthma risk.

Alice uses the blueprint “situation model” as a guide, and configures a mashup application which combines the different data streams and creates a heat-map representing asthma epidemic risk in different parts of the U.S. Continuously being updated with real-time data, the visualization gives an intuitive snapshot of the asthma situation evolving across the U.S.

Charlie sees this map on *Ushahidi.com*, and decides to take this one step further by *personalizing* the situation recognition. He defines individuals’ personal risk level based on rules that combine the individual parameters (e.g., exertion level, sneezing frequency, body temperature) for any person with the corresponding risk level in her surroundings. He defines action rules that urge the most vulnerable people to visit doctors, advise potentially vulnerable people to avoid exertion, and prompt users in healthy environments to enjoy the outdoors (e.g., “go jogging at the nearest park”). Thousands of users get this information, and based on verifying the data behind the suggestion, many decide to dust off their track pants and go for a run.

7.1.2 Difficulties in Creating Situation-Aware Applications

Despite multiple recent efforts to understand and use situational data, building an application like the one described here is often a hard and tedious process. This is due to multiple reasons, including the lack of understanding and operationalization of the concept of situations, heterogeneity of the multimodal data involved, real-time processing requirements, lack of geo-spatial data and abstractions, and a dearth of computational infrastructure support.

First, the term *situation* is used differently across application domains (see Section 7.2 for details), resulting in a lack of unified semantics and hence fragmented design efforts. Next, there is a dearth of tools to explicitly describe what the designer means by a particular situation, e.g., “Allergy Outbreak.” To recognize a concept one must 1) have an internal model of what it means, and 2) be able to externalize it using some constructs. Lack of such tools often implies that the situation models and data used in applications are driven by the acquisition mechanisms available. Further, practical problems (like the one discussed earlier) require a combination of information from different sources, which come at different spatial and temporal resolutions. This requires heterogeneous data to be converted into a common representation that is generic and does not need to be redefined for every new data source selected. For example, the discussed asthma risk application needs a method to combine the data coming from Twitter stream, satellite, and pollution neighborhoods. The data representation needs to capture the spatial semantics like neighborhood (e.g. to define the pollution effect of factories) and geography-driven joins (e.g., for overlaying of data grids).

Similarly, detecting situations involving data coming from all parts of the world requires scalable systems that can seamlessly handle huge volumes of data. Further, situations need to be recognized both at the personal level and the macro level. Traditional situation recognition has focused on single large-scale (e.g., over city, state, country) insights. The decisions once made were *broadcasted*. This was true from health warnings to weather alerts to advertisements. Today, we often need to individually access each users inputs and combine them with the surrounding situation recognized around her, thus allowing each user to get a *personalized (unicast)* alert based on a specific situation recognized for her. Lastly, with any new problem, and more crucially so when trying to reach out to application designers with variations in the level of design experience, iterative development is key to creating usable situation-aware applications. Thus there is a need for devising frameworks that support rapid iteration to fine-tune the recognition model until it fits the application requirements. This chapter summarizes a line of work that tries to tackle many (but not all) of these challenges toward building situation-aware systems.

7.2 Defining *Situation*

As mentioned in the previous section, one of the fundamental challenges in the field of situation recognition is the lack of a unified definition of *situation* as a concept. Hence, in this section we discuss the notion of *situation* as broadly understood across different research fields and then present an operational definition for it.

7.2.1 Existing Definitions

Situations have been studied across multiple research areas such as ubiquitous/pervasive computing [Yau and Liu 2006, Takata et al. 2008], building automation [Dietrich et al. 2004], mobile application software [Wang 2004], aviation/air traffic control [Endsley 1988, Adam 1993], robotics [Reiter 2001, Levesque et al. 1994], industrial control [Pospelov 1986], military command and control [Steinberg et al. 1999], surveillance [Brdiczka et al. 2006], linguistics [Barwise and Perry 1981], stock market databases [Higgins 1993, Adi and Etzion 2004], and multimodal presentation [Nazari Shirehjini 2006], under the garbs of situation modeling, situation awareness, situation calculus, situation control, and situation semantics. The interpretation of *situation*, however, is different across different areas and even across different works within the same area.

Here we sample some of the definitions employed for *situation* or *situations*:

- [Endsley 1988] “the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future”
- [Moray and Sheridan 2004] “a shorthand description for keeping track of what is going on around you in a complex, dynamic environment”
- [Adam 1993] “knowing what is going on so you can figure out what to do”
- [Jeannot et al. 2003] “what you need to know not to be surprised”
- [McCarthy et al. 1968] “A situation is a finite sequence of actions.”
- [Yau and Liu 2006] “A situation is a set of contexts in the application over a period of time that affects future system behavior.”
- [Barwise and Perry 1980] “The world consists not just of objects, or of objects, properties and relations, but of objects having properties and standing in relations to one another. And there are parts of the world, clearly recognized (although not precisely individuated) in common sense and human language. These parts of the world are called situations. Events and episodes are situations in time, scenes are visually perceived situations, changes are sequences of situations, and facts are situations enriched (or polluted) by language.”
- [Dietrich et al. 2004] “. . . extensive information about the environment to be collected from all sensors independent of their interface technology. Data is transformed into abstract symbols. A combination of symbols leads to representation of current situations . . . which can be detected”

- [Sarter and Woods 1991] “accessibility of a comprehensive and coherent situation representation which is continuously being updated in accordance with the results of recurrent situation assessments”
- [Dominguez et al. 1994] “the continuous extraction of environmental information along with integration of this information with previous knowledge to form a coherent mental picture, and the end use of that mental picture in directing further perception and anticipating future need”
- [Smith and Hancock 1995] “adaptive, externally-directed consciousness that has as its products knowledge about a dynamic task environment and directed action within that environment”
- [Dostal 2007] “the ability to maintain a constant, clear mental picture of relevant information and the tactical situation including friendly and threat situations as well as terrain”
- [Merriam-Webster 2003] “relative position or combination of circumstances at a certain moment”
- [Singh and Jain 2009b] “the set of necessary and sufficient world descriptors to decide the control output”
- [Steinberg et al. 1999] “Situation Assessment is the estimation and prediction of relations among entities, to include force structure and cross force relations, communications and perceptual influences, physical context, etc.”
- [Dousson et al. 1993] “set of event patterns and a set of constraints”

We clearly see some common traits as well as the dissimilarities amongst different definitions. Most telling perhaps is the observation by Jakobson et al. [2006] that “. . . *being a relatively new field, there is a clear lack of theoretic well-grounded common definitions, which may be useful across different domains.*”

Focusing on the commonalities rather than the differences, one may find that the following notions reverberate across definitions:

Goal based (GB). Situations need to be defined for an application or a purpose.

Space and time (ST). Situations capture and represent a volume of space and/or time.

Future actions (FA). Situations support future prediction and/or action taking.

Abstraction (AB). Situations present some form of perception or symbolic representation for higher cognitive understanding.

Table 7.1 Survey of *situation* definitions

Work	Goal Based	Space and Time	Future Actions	Abstraction	Computationally Grounded
[Endsley 1988]		X	X	X	X
[Moray and Sheridan 2004]		o		X	
[Adam 1993]	X		X		
[Jeannot et al. 2003]	X				
[McCarthy et al. 1968]			X		
[Yau and Liu 2006]	X		X		X
[Barwise and Perry 1980]		X		X	
[Dietrich et al. 2004]				X	X
[Sarter and Woods 1991]		o		X	
[Dominguez et al. 1994]	X		X	X	X
[Smith and Hancock 1995]	X	o	X	X	
[Dostal 2007]		o		X	
[Merriam-Webster 2003]		o			
[Singh and Jain 2009b]	X		X		X
[Steinberg et al. 1999]	X		X	X	o
[Dousson et al. 1993]		o	X	o	X

Note: 'o' indicates partial support.

Further, while some definitions are *computationally grounded (CG)* in data (e.g., Endsley, Dietrich), others were abstract (e.g., Barwise and Perry, Merriam-Webster). (A summary of these definitions based on the abovementioned axes is presented in the Table 7.1.)

7.2.2 Proposed Definition

Based on observing these common traits (as well as a focus on staying computationally grounded), we define a *situation* as:

An actionable abstraction of observed spatio-temporal descriptors.

Going right to left, let us consider each of the terms used in this definition:

descriptors. This follows the approach of quantifying an abstract/inexact notion based on sampling its characteristics [Duda and Hart 1996, Nowak et al. 2006].

spatio-temporal. The most common connotation associated with *situations* (as well as this work’s focus) is spatio-temporal data.

observed. As a computational concept, the focus is only on the “observable” part of the world. Meta-physical as well as physical aspects that cannot be measured by sensors are simply beyond its scope.

abstraction. This signifies the need to represent information at a much higher level than sensor measurements or even their lower-level derivations. Decision-makers typically focus on higher- (knowledge-) level abstractions while ignoring the lower-level details.

actionable. The top-level descriptors and abstractions need to be chosen based on the application domain, and the associated output state-space. Hence our focus is on creating a representation (e.g., classification) that maps the lower-level details into one concrete output decision descriptor. Hence, we are not interested in *any* higher-level abstraction, but rather the *specific* one that supports decision-making in the application considered.

As can be noticed, this definition operationalizes the reverberating threads found across different definitions in literature, and computationally grounds them.

7.2.3 Problem of Situation Recognition

As highlighted by the definition, the essential problem of situation recognition is that of obtaining actionable insights from observed spatio-temporal data. Just like any effort at concept recognition, this problem can be split into phases: observing data, extracting features, and detecting concepts from the observed features.

The unique nature of the situation recognition problem is reflected in the spatio-temporal grounding of all data, as well as of the features defined.

7.2.4 Data

Let us represent the observed data at spatio-temporal coordinate st about any particular theme θ as follows:

$$D_{st\theta} = \lambda(\theta, st), \quad (7.1)$$

where:

s represents the spatial coordinate of the observation, i.e., $s \in \mathfrak{R}^3$,

t represents the temporal coordinate of the observation,

θ represents the application-/sensor-specific properties that are observed at the spatio-temporal coordinates, and λ is the mapping function from the real-world characteristics to the observation space.

Aggregating over space and time, the data about any particular theme can be referred to as $D_{ST\theta}$, and combining over all observed themes, the data D_{ST} can be represented as:

$$D_{ST} = \{D_{ST\theta_1}, D_{ST\theta_2}, \dots, D_{ST\theta_k}\}. \quad (7.2)$$

7.2.5 Features

A spatio-temporal feature f_{ST} can be obtained via a function Ω applied on the observed data:

$$f_{ST} = \Omega(D_{ST}). \quad (7.3)$$

These features (e.g., growth rates, geographical epicenters, raw values) capture different properties of the observed phenomena and are selected based on their ability to discriminate between the classes of interest.

The combination of features yields a feature set F_{ST} represented as:

$$F_{ST} = \{f_{ST1}, f_{ST2}, \dots, f_{STN}\}. \quad (7.4)$$

7.2.6 Situations

Consequently, situations can be derived via a function Ψ applied on the feature set.

$$c_{ST} = \Psi(F_{ST}). \quad (7.5)$$

and $c_{ST} \in C$, where C is the situation universal set. It could be discrete classifications (the focus of this work) or values in a certain range. Here:

$$C = \{c_1, c_2, \dots, c_m\}, \quad (7.6)$$

where c_1 through c_m are the admissible classes of situation.

To summarize, c_{ST} is the final spatio-temporal situation selected from the range of situations possible, obtained via function Ψ applied on the observed features, which in turn are obtained by applying a function Ω on the observed spatio-temporal data.

Thus the problem of situation recognition is to identify the right situation classification for a given set of observations, i.e.,

$$\Psi \circ \Omega : D_{ST} \rightarrow C, \quad (7.7)$$

or alternatively:

$$c = \Psi(\Omega(D_{ST})). \quad (7.8)$$

The goal of the current discussion is to define a framework to tackle the situation recognition problem, i.e., extract spatio-temporal features from the observed data, and use them for situation classification. Note that in the current discussion we will focus on two dimensions (latitudes and longitudes) for spatial coordinates, i.e., $s \in \mathfrak{R}^2$, and consider the observed values to be real numbers, i.e., $D_{st\theta} \in \mathfrak{R}$.

7.3 A Framework for Situation Recognition

An important pathway to progress in the field of situation recognition is to create a generic framework that allows for the creation of multiple situation-aware applications. Based on the analysis of multiple existing situation-aware applications (see [Singh and Jain \[2016\]](#), Chapter 2, for details), here we discuss three design goals for such a framework.

1. Expressive power
2. Lower the floor
 - (a) Reduced time to build
 - (b) Lower computer science (CS) expertise required
3. Raise the ceiling
 - (a) Better designed situation detectors
 - (b) Personalization options provided

The concepts “lower the floor” and “raise the ceiling” are inspired by [Myers et al. \[2000\]](#).

Expressive power. Given the diversity observed in various applications, a framework designed to recognize situations across them needs to be versatile. This implies that the framework needs to focus on the commonalities, and also start with aspects that are common across applications. The last-mile specific issues can be left to the individual application designers where required.

Lower the floor. The framework should resonate with the ideals of [Carrier et al. \[2008\]](#) in that this “. . . new breed of applications, often developed by nonprofessional programmers in an iterative and collaborative way, shortens the traditional development process of edit, compile, test, and run. Situational applications are seldom developed from scratch; rather, they are assembled from existing building blocks.” To truly allow web-scale innovation, and cater to the “Long tail of (situation) applications” [[Viedma 2010](#)], the framework needs to make sure that the situation detectors are easy to build and do not presume computer science (CS) expertise. The user input needs to be a declarative specification of *what*. The procedural details of *how* need to be abstracted away wherever possible.

Raise the ceiling. The framework should not only support situation recognition but also raise the quality of the detectors defined. We consider two different aspects of this raising of the ceiling. First is the design process of the applications. The framework should include design guidelines and wizards to ensure that the designers do not fall into common early mistakes. This means that the complexities of operator implementation or writing data wrappers should no longer be a factor in influencing which affordances are provided by an app. Once the design process selects certain modules, they should be available at minimal cost. Second is the ability to support personalization. Traditional situation recognition and decision-making has focused on single large-scale (e.g., over city, state, country) decision-making. Today, we need tools to individually access each user’s inputs and combine these with the surrounding situation for personalized decision-making.

7.3.1 Components Required for the Framework

In order to support the design goals discussed, three important components are required for such a framework.

The building blocks. To increase the expressive power and to “lower the floor,” we need to identify common building blocks that can be used to build a wide variety of applications. The building blocks need to be built upon abstractions that are commonly understood by all application developers, and are applicable across different applications (e.g., space and time). In the presented framework, situation recognition operators (see [Section 7.3.2.5](#)) provide such building blocks.

Modeling approach. To “lower the floor” the framework needs to provide a set of guidelines so that a new application designer is not overwhelmed by the task at hand. Building a system to handle the “Asthma Risk across U.S.” might appear to be too vague and daunting for a new user. But with some guidance through the

design process, a user can break a problem into modular, explicit, and computable chunks. Equally importantly, the resulting guidelines can help “raise the ceiling.” The framework addresses this aspect by defining a step-by-step process that breaks down complex situation representations into smaller (easier to define) concepts until they can be evaluated using a single data stream, using a single operator. See [Singh and Jain \[2016\]](#) Chapter 5, for more details.

Rapid prototyping toolkit. The end goal of the framework is to build working applications for personal and societal good. Hence providing a toolkit (graphical or API based), which allows the users to quickly translate the models into working applications, will help “lower the floor.” On the other hand, an ability to rapidly re-iterate and redefine the applications will help “raise the ceiling.” Further, an ability to personalize the recognized situations and configure action alerts will help raise the ceiling. EventShop (discussed in more detail in Section 7.4) is one such rapid prototyping toolkit that has been developed to support the framework.

7.3.2 Situation Recognition Workflow

The process of moving from heterogeneous streams to situations involves 5 steps:

1. Stream selection
2. Data ingestion
3. Data unification
4. Aggregation
5. Situation evaluation

Relevant data streams are identified by the domain experts, based on which the relevant wrappers ingest the data. A unified spatio-temporal format records the data originating from any spatio-temporal coordinate using its numeric value. Aggregating such data results in two-dimensional data grids (called E-mages, described in more detail later), which can be extended over time into E-mage streams. The situational descriptor is defined as a function of different spatio-temporal features derived using operations applied on the E-mage streams.

7.3.2.1 Data stream selection

The situation models act as blueprints that identify the data streams and operators required to recognize a situation. The data streams can originate from a person’s mobile device, social network updates, stand-alone sensors, satellites, websites, or

archived web data sources. Conceptually, as many different types of raw data as may be relevant for situation recognition can be selected.

7.3.2.2 Data ingestion

The ingestion of data from different sources is based on wrappers that bring the external data into the system. For computational purposes, all data streams are normalized to numeric streams, but the underlying “raw” data is also maintained while required by the application. Each data stream has a set of associated spatio-temporal parameters that need to be configured.

7.3.2.3 Data unification

The heterogeneous data streams are unified based on focusing on the commonalities across them. Each data stream is considered to be reporting certain thematic observations from different spatio-temporal coordinates. Hence an STT or space, time, theme (i.e., where-when-what) tuple is used to organize all types of data.

An STTPoint is represented as:

$$STTPoint = \langle latitude, longitude, timeStamp, theme, value \rangle \quad (7.9)$$

By extension, a flow of STTPoints becomes an STT stream.

7.3.2.4 Spatiotemporal aggregation

Spatial data can be naturally represented in the form of spatial grids with thematic attributes. Values in STTPoints that are collected in a time window over an STT stream can be combined and aggregated to form a two-dimensional data grid. The data grid together with related STT information is called an E-mage. E-mages capture the semantics and notion of a spatial neighborhood very elegantly, and geographical joins [Hjaltason and Samet 1998] between data streams reduce to simple overlaying of grids. An example of an E-mage is shown in Figure 7.3.

A flow of E-mages forms an E-mage stream, which serves as a first-class citizen, i.e., a fundamental data structure in the framework.

These gridded representations of spatio-temporal data are very analogous to images, pixels, and videos, which have been studied by media processing researchers for a long time. These analogies are relevant from multiple perspectives.

Visualization. The grid-like representation allows for intuitive visualization and aids situation awareness for a human user. Humans are quite used to seeing satellite image and geographical information systems (GIS) data on similar interfaces.

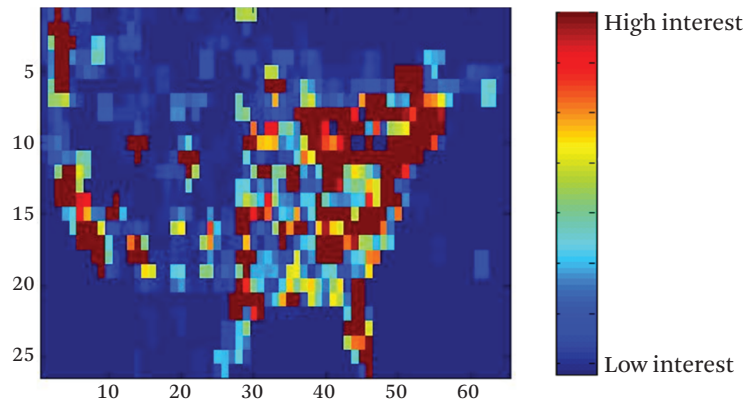


Figure 7.3 An E-mage showing user interest across the mainland U.S. in terms of the number of tweets containing the term iPhone on 11th Jun 2009. (From Singh et al. [2010e])

Intuitive query and mental model. The correspondence of the human mental model of spatio-temporal data with the query processing model makes it easier for humans to pose queries, and understand the results.

Data analysis. Such a representation allows for exploitation of a rich repository of media processing algorithms that can be used to obtain relevant situational information from this data. For example, well-developed processing techniques (e.g., filtering, convolution, background subtraction) exist for obtaining relevant data characteristics in real time.

Efficiency. The pixel/image-based representation reduces the run-time processing requirements from potentially millions of XML data feeds to a rectangular grid representation of known size. This allows the *run time* query process to work on just the E-mages (which can be directly stored in the main memory due to much smaller size), rather than the entire raw data corpus. The process of creating E-mages out of the raw data can be undertaken separately without affecting run-time performance.

Privacy preservation. Such an aggregation approach aids applications that need to maintain *individual* user privacy. Spatio-temporal “binning” allows the higher-level algorithms to work on the aggregate representations, without focusing on individuals.

7.3.2.5 Situation evaluation

The situation at any location is characterized based on spatio-temporal descriptors determined by applying appropriate operators on E-mage streams. We identify a

	Operator type	Input E-mage stream	Supporting parameter(s)	Output
Π	Filter		+ Mask	
\oplus	Aggregate		+	
γ	Classification		Classification method	
@	Characterization		Property required	
Ψ	Pattern matching		+ Pattern	

Figure 7.4 Different operators for situation recognition. (Adapted from Singh and Jain [2016])

core set of operators for situation evaluation: *Filter*, *Grouping*, *Aggregation*, *Spatio-temporal Characterization*, and *Spatio-temporal Pattern Matching*. These operators are designed to be declarative to allow end users to describe their data needs rather than procedurally handling the details of manipulating the data. The aim of designing these operators is to retrieve relevant spatio-temporal-thematical data (E-mages, E-mage streams, or their attributes) by describing their characteristics [Ullman 1983].

An overview of the operators is shown in Figure 7.4.

While interested readers are referred to Singh and Jain [2016] for more formal definitions of the operators, they are summarized as follows:

Filter (Π). For selection of data based on space, time, theme, or value parameters.

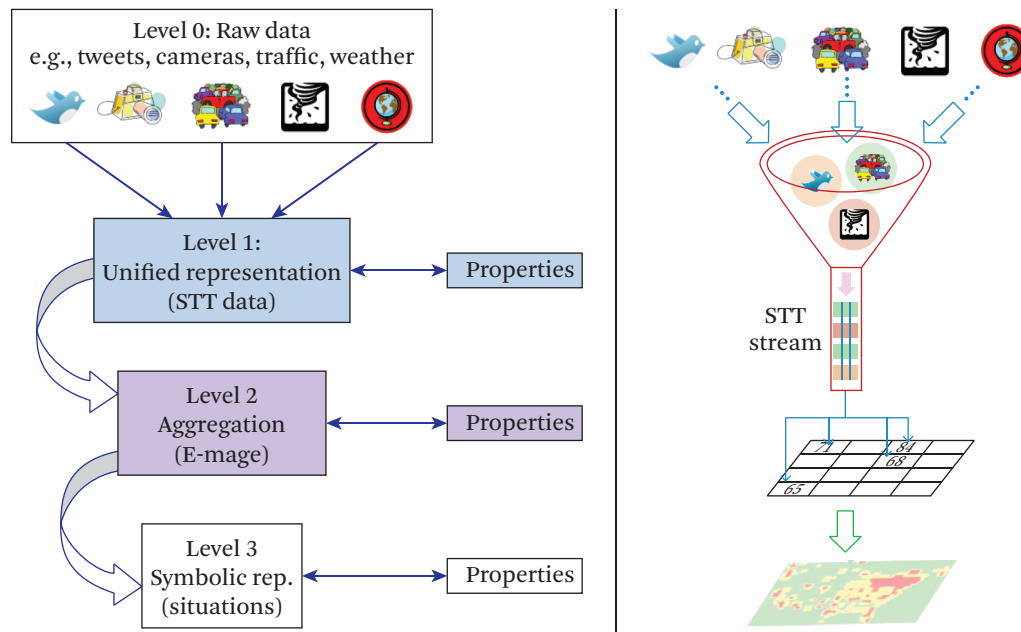


Figure 7.5 Workflow for situation recognition. (From Singh et al. [2012])

Aggregate (\oplus). To combine multiple E-mage streams based on mathematical operators.

Classification (γ). To segment the values into different categories (e.g., high, mid, low).

Characterization [spatio-temporal] (@). To derive different spatio-temporal attributes (e.g., epicenter, growth-rate, shape) of any E-mage stream.

Pattern Matching [spatio-temporal] (Ψ). To compare data with known patterns or historical spatio-temporal data.

Heterogeneous sensor streams can be combined to derive actionable situations as shown in Figure 7.5. The unified STT format employed (level 1) records the data originating from any spatio-temporal bounding box using its numeric value. Aggregating such data results in two-dimensional data grids (level 2). At each level the data can also be characterized for analytics. The situational descriptor (level 3) is defined by the user (application expert) as a function of different spatio-temporal characteristics.

7.3.3 Responding to Situations

The evaluated situation information can be used for visualization, analysis, and control actions. Spatio-temporally aligned situation data can be visualized on maps and timelines, and reported as text. The outputs of different queries are either a temporal stream of E-mages (which can be overlaid on maps), or STTPoints (which can be shown on a map and a timeline), depending on the type of query.

The recognized situation can be personalized to each user by projecting out the situation classification value for the user's location and combining it with personal parameters. The individual parameters can be personal data streams (e.g., activity level) and action recommendations can be undertaken using approaches similar to ECA (event-condition-action) [Montanari et al. 2007]. The spatio-temporal coordinates associated with each data stream are used to direct users to the nearest location satisfying certain conditions. A combination of macro-situations and personal parameters can be used to configure different situation-action templates. Multiple such templates can be registered to provide customized alerts to all recipients.

7.4 EventShop: A Toolkit for Situation Recognition

Based on this framework, we have developed a web-based platform called EventShop [Singh and Jain 2016, Gao et al. 2012, Pongpaichet et al. 2013] (<http://auge.ics.uci.edu/eventshop/>¹) that provides an easy way for different users to experiment with different data streams and recognize situations. This system operationalizes the various concepts promulgated in the framework. It provides an easy way to test and refine situation models, and does so using the data representation and operation algebra.

EventShop provides operators for data stream ingestion, visualization, integration, situation characterization, and sending out alerts. The system can be graphically configured to interactively recognize different situations and undertake corresponding actions. It adopts a modular approach to make the system reconfigurable for different applications “on the fly.” A simple graphical interface makes it accessible to non-technical users.² Hence, for the first time it provides non-technical users an opportunity to experiment with real-time data streams coming from all parts of

1. EventShop is an evolving open-source project. Source code for the current version is available at <http://github.com/Eventshop>.

2. The intended users are application designers. They need not be computer-science experts, but they are expected to have access to the application logic.

the world and integrate them for diverse applications, thus making one concrete step toward democratization of the process of situation-driven app-building.

EventShop includes a front-end user interface (UI) and a back-end stream processing engine. EventShop draws inspiration from PhotoShop and provides an environment that allows users to apply different filters and operators to experiment with multiple layers of data until they are satisfied with the processing result. Just like PhotoShop moved image processing from *specialists* to the *common-person* domain, EventShop aims to make real-time data processing and action-taking capabilities easy and available to all. EventShop is designed to allow its users to experiment with data sources and formulate queries by combining a rich set of operators without worrying about the underlying technical details.

A screenshot from EventShop is shown in Figure 7.6. The basic components are:

Data-source Panel. To register different data sources into the system.

Operators Panel. To show the different operators that can be applied to any of the data streams.

Intermediate Query Panel. To display a textual representation of the intermediate query currently being composed by the user.

Registered Queries. To list completed queries registered with the system.

Results Panel. To show the output of the query (which can be presented on a map or timeline, as a numeric value, or as combination of these).

7.4.1 System Design

EventShop provides a graphical interface that allows end users to register new data stream sources and formulate queries by combining a rich set of built-in operators. Users are also provided with a graphical user interface (GUI) tool that allows them to send personalized alerts to relevant people. On the back-end, data sources and queries requested from the front-end are stored into data source and query databases. Based on the information of registered data sources, EventShop continuously ingests spatio-temporal-thematic data streams and converts them to E-mage streams. In the meantime, directed by the registered queries, EventShop pulls E-mage streams from data ingestors into the query processor, which then processes the E-mage streams in each of the instantiated query operators. Besides being converted to E-mage streams, the raw data stream (e.g., tweet stream) is also persisted into raw data storage. This raw data can be combined with situation query results to define action conditions in the Personalized Alert Unit.

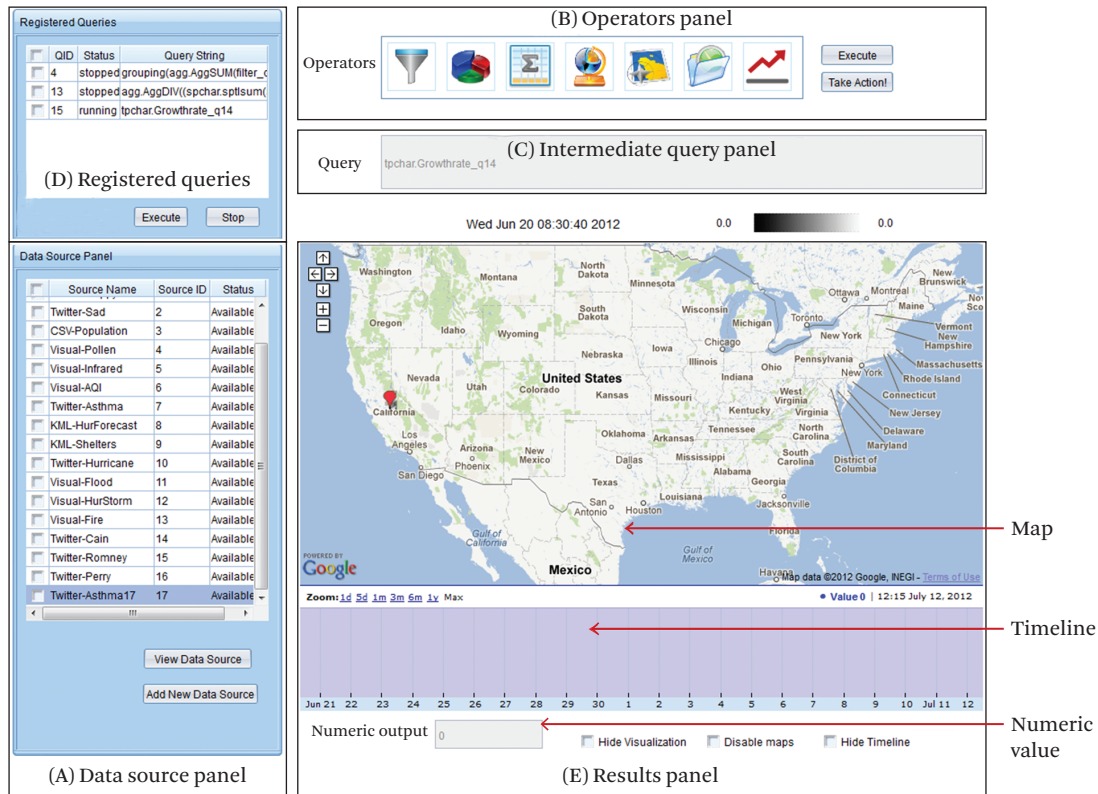


Figure 7.6 Screenshot of EventShop system.

The front-end GUI of EventShop is implemented in JavaScript, and sends requests to the back-end controller through a set of Ajax calls. The back-end controller to respond to these requests is implemented using Java Servlets. The data ingestor component is implemented in Java. The implementation of runtime operators makes use of the OpenCV package [Bradski and Kaehler 2008] and is written in C++.

The system architecture is shown in Figure 7.7. There are four main components: Data Ingestor, Query Processing Engine, Internal Storage, and Action Control. After a data source is registered, and parsed to the Data Ingestor, a new data adapter is created to connect to the data source; it takes the raw spatio-temporal-thematic data stream as input, and relies on an E-mage generator iteration to convert the raw data stream into an E-mage stream. Specified by the users, raw data streams and/or E-mage streams can be stored in the Internal Storage. After

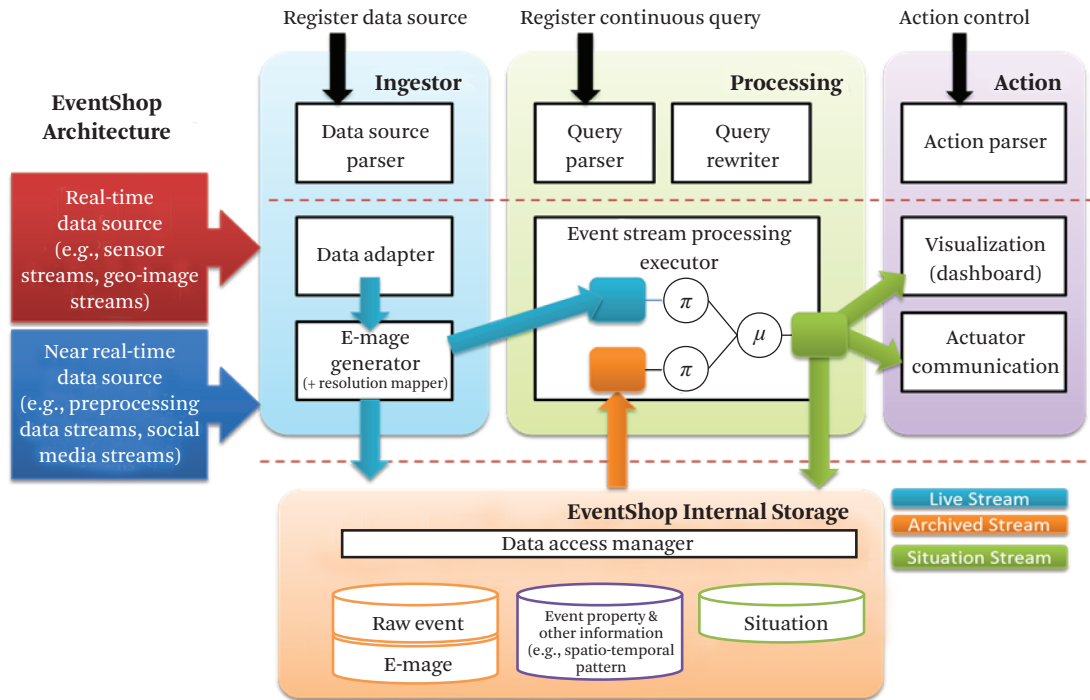


Figure 7.7 System architecture of EventShop.

registered data sources, the situation recognition model can be formed by applying different spatio-temporal operators on any of the data sources, and registered via the Query Processing Engine. This situation model is represented as a rooted query plan tree (logical operator tree) to show an ordered set of steps for accessing or processing E-mages. Nodes of a query plan tree are either Data Access Node (DAN) or Operation Node (OpNode). Leaf nodes of a query plan tree are always DAN, and internal nodes are OpNode. The root node of a query plan tree represents the last operator in the plan applied to an E-mage. Edges between nodes represent the E-mage stream flowing from a lower-level node to a higher-level node. Again, the output situation streams may be stored for future use. Finally, Action Control allows application developers or domain expert analysts to visualize the output via the dashboard, and to provide a quick response to the end users.

7.4.2 Ingesting Heterogeneous Data

EventShop combines different data streams by focusing on the commonality between them. The fundamental data structure used for combining spatio-temporal data is an STTPoint:

$$STTPoint = \langle Latitude, Longitude, TimeStamp, Theme, Value \rangle.$$

As can be seen, an STTPoint can be defined out of very disparate data sources, ranging from a traffic sensor (theme = road speed) to a twitter feed (e.g., theme = allergy incidents). Values in STTPoints that are collected in a time window over the STT stream can be combined and aggregated to form a two-dimensional data grid. The data grid, together with related STT information, is called an E-mage, represented as:

$$E\text{-mage} = \langle Bounding\ Box\ Coordinates, Latitude\ Resolution, \\ Longitude\ Resolution, Time\ Stamp, Theme, 2D\ Data\ Grid \rangle,$$

where the two-dimensional data grid captures the observed values at each cell and the supporting meta-data uniquely maps those values from the grid representation to the real-world spatio-temporal coordinates. Examples of multiple E-mages (overlaid onto a map) can be seen in Figure 7.8. The use of a grid is based on the understanding that grids are the fundamental data structure used by humans to understand and analyze spatial data (e.g., maps, satellite images).

Data ingestion from different sources is undertaken using wrappers, which translate the external data into E-mage streams. The types of data streams supported in the system will evolve as it is used for diverse applications. For computational purposes, all data streams are normalized to numeric streams. Each of the data streams has a set of associated spatio-temporal parameters, which need to be configured. EventShop assumes that the application designers will identify the data sources that match the quality and relevance requirements of their applications. The following wrappers are currently available in EventShop to support data ingestion from a variety of data sources and formats:

External systems APIs. For example, Twitter streams, Flickr streams, Event-Warehouse streams developed by NICT (Japan)

Web geo-images. For example, air pollution level map of U.S. from www.epa.gov, and satellite images

Web structured data. From valid URL (in KML or specified CSV format)

Mobile phone sensors. Using “Funf-in-a-box,” <http://funf.media.mit.edu/>

Archived data. From MySQL and MongoDB database

7.4.2.1 Situation Recognition Operators

EventShop offers an implementation of the five basic situation recognition operators described in Section 7.3.2.5: *Filter, Grouping, Aggregation, Spatio-temporal*

Characterization, and *Spatio-temporal Pattern Matching*. It also supports two additional operators:

Conversion. To convert data at different granularity into a desired granularity over space-time dimensions (e.g., convert coarse granularity E-mage to fine granularity using equal split method).

Interpolation. To handle sparse data points, this operator allow users to interpolate unknown data using nearby values.

These operators are designed to be declarative rather than procedural [10]; i.e., this algebra is designed to retrieve relevant spatio-temporal data (E-mages, E-mage streams, or their attributes) by describing their characteristics, rather than manipulating them directly. Lastly, each of these operator types also has a set of configurable parameters that define the exact output (e.g., “Aggregate” can be used to subtract, sum, multiply, XOR, or convolve two streams; “Characterization” can be used to obtain epicenter, growth-rate, velocity, and so on).

7.4.2.2 Visualization of the Data and the Results

Users can click on any of the data sources to visualize them overlaid on the map and configure them based on application needs. Similarly, users can see the results of various queries in the results panel. Based on the type of query, results are shown on a map, timeline, a numeric textbox, or a combination of those. To support easier visualization, any visualized E-mage (in gray-scale) is scaled to the browser-friendly range of 0 to 255, and the results of the *Classification* operation are color-coded (e.g., Red, Green, Yellow) to illustrate different categories.

7.4.2.3 Personalization and Alerts

Personalized situations are detected by combining the detected macro situation at the user’s location with her personal parameters. The detected personalized situations can be used to trigger different alerts. EventShop works on E-C-A (event-condition-action) [Montanari et al. 2007] templates where users with certain personal conditions located in an area with a prescribed situation can be directed to the nearest location with desired conditions via alert messages. Multiple such E-C-A templates are registered with the system to provide customized alerts to all recipients.

7.5 Building Situation-Aware Applications Using EventShop

EventShop and the associated situation recognition framework has so far been used for applications like hurricane detection and mitigation, identifying weather pat-

terns (e.g., fall colors in New England), showing influence patterns for different political figures, identifying demand hot-spots for business products, displaying allergy risk and recommendations, detecting flu outbreaks and wildfires, evaluating Global Warming Index, Quality of Living Index, and flood mitigation [Singh et al. 2010e, Singh et al. 2012, Gao 2012, Singh and Jain 2016]. Here, we expand on the basics of the example in Section 7.1, to illustrate the specific process of creating one representative application using the framework.

7.5.1 Asthma/Allergy Risk Recommendation

Asthma affects more than 300 million people worldwide. Here we discuss the creation of an actual app to detect the asthma/allergy risk level across the U.S., and advise the highly vulnerable people to stay indoors, while prompting those in healthy environments to go out jogging. On an experimental basis, this time we define the asthma/allergy risk for an environment based on the pollen count, air quality, and the number of asthma reports on social media (human sensors) in the neighborhood. For this, one may use the data source panel of EventShop and configure the parameters <Theme, Resource Location, Type, Time Window, Time Synchronization Point, Spatial Bounding Box, Spatial Resolution> for the three sources as follows:

1. <Pollen Count, http://pollen.com/images/usa_map.gif, Geo Image Stream, 1 day, 0 ms, USA, 0.1 Lat x 0.1 Long>
2. <Air Quality, <http://airnow.gov>, Geo Image Stream, 1 day, 0 ms, USA, 0.1 Lat x 0.1 Long>
3. <Asthma Tweets, (Twitter Search API URL), Text Stream, 6 hours, 0 ms, USA, 0.1 Lat x 0.1 Long>

As shown in Figure 7.8, these operators can normalize the stream values to the same range, aggregate them using a “sum” function, and classify the output into three levels representing high, mid, and low level of asthma/allergy risk. Working on real-time data, this query will show different results based on the incoming data at that time. Thus without coding or expert-level experience, one may start observing an experimental real-time Asthma/Allergy Risk Level map across the whole of the U.S. One may use this visualization for drawing insights from it, or choose to quickly modify this to experiment with other data streams. One may also extend it to support personalized alerts. For example, there are multiple ongoing efforts to support easy capture and analysis of real-time, personal data streams. Funf-in-a-box (<http://funf.media.mit.edu/>) allows users to create mobile sensing apps without any programming. We have built one such mobile app called RelaxMinder and placed

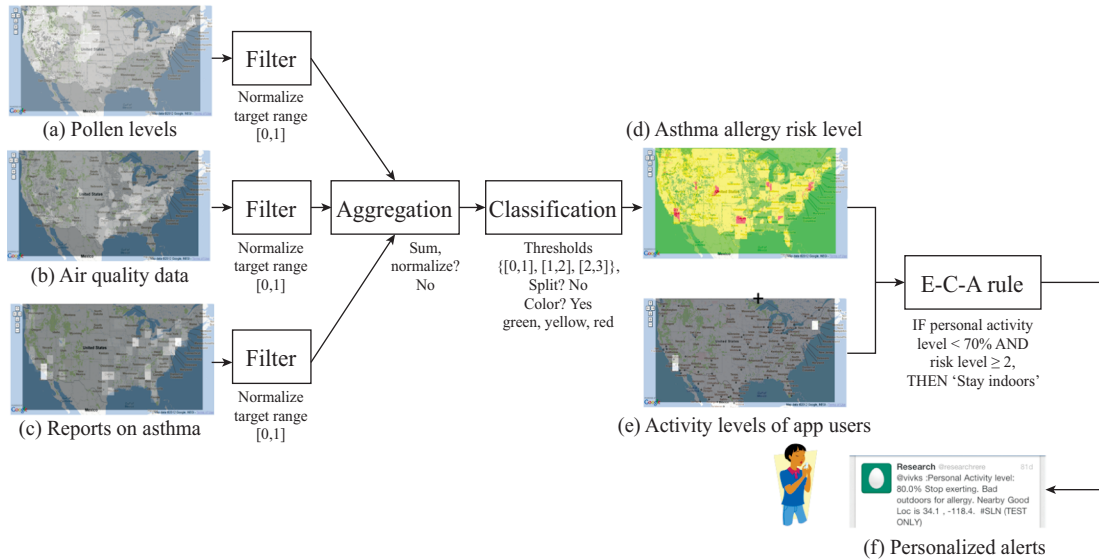


Figure 7.8 Creating Asthma/Allergy Risk Level maps by combining different data streams in EventShop.

it on the Android marketplace. Multiple interested users have installed the app, and this provides a stream of their physical activity levels and geo-coordinates. An interested user can configure this data source in the source panel with the following parameters:

4. (Personal Activity Level, (Secure RelaxMinder Stream URL), Text Stream, 6 hours, 0 ms, USA, 0.1 Lat x 0.1 Long)

Now the personalization alert unit can be configured with rules, which combine the (already detected) macro situation with the personal activity level values to send out personalized alerts as tweets to registered users (e.g., IF personal activity level < 70% AND Risk Level ≥ 2, THEN 'Stay Indoors').

The pilot run for the asthma/allergy mitigation application involved installation and usage by seven users who were explicitly contacted by the research group. Based on the feedback by these users, a UCIAsthma Android application has been developed and released on the Android market.

While here we limit the discussion to the creation of one situation-aware application, the same situation-recognition framework has also been applied to multiple applications including wildfire recognition, flu monitoring and alerting, business decision-making, flood alerts, seasonal characteristics analysis, and hurricane

monitoring [Gao et al. 2012, Singh and Jain 2009b, Singh et al. 2009a, 2010b, 2010c, 2010d, 2010e, 2012]. The ability of the framework to support creation of diverse situation-aware applications, each with sophisticated situation-recognition models, while still needing minimal CS expertise from the app developers, suggests that the framework has made a useful stride toward its goal of “lowering the floor” and “raising the ceiling.” Interested readers are referred to Singh and Jain [2016], Chapter 9, for a more detailed discussion.

7.6 Open Challenges and Opportunities

While the presented line of work has made some early inroads into understanding and building situation aware applications, these efforts still mark early days in this research direction. Multiple research challenges remain, and for those, the multimedia research community is ideally suited to contribute.

7.6.1 Challenge 1: Handling Missing and Uncertain Data

Data remains the underlying foundation on which situation-aware systems need to be designed. Incoming data streams in such applications may have associated uncertainty with them, which, if ignored, can lead to imprecise or invalid conclusions. The data structures also need to support data representation with uncertainty and define relevant spatio-temporal operations. Researchers in the statistics, pattern recognition, machine learning, and multimedia research communities often represent uncertain data as vectors. Univariate and multivariate distributions have been defined to model different types of uncertainties. In many applications, owing to the central limit theorem [Kallenberg 2006], a Gaussian distribution is a reasonable distribution to model uncertainties like measurement errors or spatio-temporal correlations in data. Operations like sampling, evaluation, conditioning, and marginalization have been well defined for Gaussian distributions. Similarly, matrix transforms on multivariate Gaussians have well-defined analytical forms.

One approach could be to bring these well-established concepts from the field of pattern recognition and adapt them in a spatio-temporal data processing framework. For example, one could define a distribution E-mage or dE-mage to represent a spatio-temporal Gaussian distribution. The dE-mage uses an E-mage to represent the mean of the Gaussian, and a positive-definite symmetric matrix to represent the covariance structure. Linear spatio-temporal operations like sum, mean, weighted mean, etc. can be expressed as matrix transformations on the underlying multivariate Gaussian distribution [Harville 1998]. Recent efforts like Wang and Kankanhalli [2015], and Wang et al. [2016d] discuss the probabilistic representation of E-mages

to handle uncertainty in the data streams. Similarly, to handle missing data, different spatio-temporal interpolation techniques need to be explored. [Tang et al. \[2015\]](#) describes one early effort in this direction.

7.6.2 Challenge 2: Supporting Evolving Situation Estimation

While the current situation recognition framework focuses on recognizing current situations, an ability to estimate future situations has applications in multiple domains. There are several well-established models for prediction including Auto-Regressive, Moving Average, and their derivatives. Models rich in capturing semantics of space are usually found to be better than models that do not account for correlations between nearby spatial measurements [[Shekhar et al. 2002](#)]. Space-Time ARIMA (STARIMA) models have been found to be effective at capturing spatial correlations as well as being economical in the number of parameters to be learned, as compared to fully descriptive models like Vector ARMA (VARMA) [[Kamarianakis and Prastacos 2003](#)]. Progress toward defining situation-aware systems would require support for spatio-temporally varying situations. This requires: (1) modeling the spatio-temporal data using a unified format, and (2) use of composite spatio-temporal representations for prediction. Past work has shown the successful use of grids (e.g., E-mages and adjacency matrices) for representing spatio-temporal phenomena. Similarly, while the Kalman filtering approach is often used for linear systems, a semantic version (cognizant of space and time) is required for supporting evolving situations in the emerging context.

7.6.3 Challenge 3: Modeling Individual Behavior for Personalized Situation Recommendations

Multiple sensing platforms (e.g., Funf) focus on obtaining and providing a variety of low-level multimodal signals (e.g., accelerometer, temperature, location) that can be used to model the individual. Such modeling can capture spatio-temporal coordinates and current exertion level, but also higher-level personal descriptors including personality traits [[de Montjoye et al. 2014b](#)], propensities to trust others [[Shmueli et al. 2014](#)], tendency to cooperate [[Singh and Agarwal 2016](#)], and so on. Creation of these higher-level inferences requires a translation of the low-level sensor data into higher-level events, concepts, and behavioral patterns (e.g., daily walk, exertion level, affect). Identifying such behavior allows the application designers to deal with higher-/ more semantic-level constructs to define the application logic, and also aids privacy by allowing the end user to provide generalized information rather than precise raw signal values (e.g., sharing that the user walked a mile, rather than the precise geo-coordinate trajectory).

Techniques in literature have studied both the abstraction of the sensors and the behaviors. For example, virtual sensors can be defined on a middleware platform, which combines values coming from multiple sensors via rule-based descriptions [Girolami et al. 2008, Gellersen et al. 2002]. Human behavior can also be abstracted by considering it to include certain patterns [Magnusson 2000, Gonzalez et al. 2008], which can be identified via techniques like EigenBehaviors [Eagle and Pentland 2006], semi-latent topic models [Wang and Mori 2009], or state-transition models [Isoda et al. 2004]. The earlier work on EigenBehaviors [Eagle and Pentland 2006] has shown that human mobility behavior can be defined on the basis of a few principal behavioral components. Further, just a subset of them is enough to derive higher-level concepts like organizational affiliation. Future efforts in multimedia research need to expand on such approaches to cater to all kinds of quantifiable behaviors and consider the real-time requirements of situation-aware systems.

7.6.4 Challenge 4: Multimodal Interfaces for Persuading User Action

Existing efforts have focused on either detecting evolving situations or independently on understanding human behavior. There has been relatively little work that combines the strengths of the two well-developed fields for persuading humans to take actions and change behaviors based on an accurate understanding of the user, her social network, and the evolving situation. This is important for multiple practical applications. For example (as illustrated earlier), a user with allergies could be nudged to exercise on a pollen-free day. Similarly, such a system could detect when a person is about to relapse and start smoking again and alert a roommate to talk to the person and create social interventions [Singh et al. 2014]. Similarly, such a system could play the (recorded audio) health advice from one's daughter when one is about to order fast food, or advise one to call their parent who has been detected to be lonely. All of these scenarios require a multimodal understanding of the user, her situation, and her social network. There is currently a dearth of such systems that can traverse multiple channels, multiple modalities, and multiple disciplines to build human-centric solutions. With the growth of multimodal content in social networks and the ease of access to devices that can both capture and present multimodal information, we would expect multimodal interfaces to play a significant role in supporting persuasion and behavior change.

The photograph, in particular, has long been perceived to have a special power of persuasion, grounded both in the lifelike quality of its representation and in its claim to mechanical objectivity [Barthes 1981]. Photos grab attention and evoke direct emotional response [Messaris 1996]. Similarly, music and lighting are known to cause a change in consumer behavior [Gorn 1982]. With the growth in virtual

reality, augmented reality, projection, display, and multi-sensory systems, it is only to be expected that these modalities will be used to create compelling and persuasive narratives for users and encourage them to undertake behaviors identified by situation-aware systems. Several early examples of these approaches are already in practice. For example, [Fortmann et al. \[2016\]](#) have defined CubeLendar, which aims to notify users about calendar events via light and represent potential situations for spontaneous communication with remote co-workers. Similarly, [El Ali et al. \[2016\]](#) give users of e-cigarettes color-coded feedback to encourage behavior change. Lastly, [Matviienko et al. \[2016\]](#) have suggested the use of ambient light displays for turn-by-turn navigation in cars. However, there remain many more opportunities for multimedia community to design and develop compelling multi-sensory experiences for persuasion and behavior change.

7.6.5 Challenge 5: Privacy and Ethical Issues

When designing societal-scale cyber-physical systems, it is important that we remember that systems are designed for human benefit (and not the other way around). This implies that users' control of their personal data and right to privacy are of critical importance. Many such issues become even more complicated with the growth of multi-sensory, multimodal social data. For example, does the leakage of one's siblings' pictures (or genomic data) count as a privacy loss for a person? Similarly, the implications of geo-temporally inscribed data and its long-term release are only partially understood (e.g., [De Montjoye et al. \[2015\]](#), [Nouh et al. \[2014\]](#)). Some of the early efforts in this direction include projects like OpenPDS and Data Box [[de Montjoye et al. 2014a](#), [Haddadi et al. 2015](#)]. Similarly, recent efforts have tried to automatically infer users' privacy needs based on their phone-use behavior [[Ghosh and Singh 2016](#)]. However, the role of multimodal data, its fusion to derive newer meaning, and its implications are not fully understood as yet.

Besides these, there remain multiple other challenges including the issues of scalability, battery life, user interaction design, societal-scale optimization, and incentive mechanisms, but we leave them outside the scope of the current discussion. In summary, there exist multiple challenges and opportunities toward designing generic frameworks that support situation-aware computing.

7.7 Conclusion

Situation recognition is an important technical and societal problem with relevance to the multimedia research community. This chapter has summarized a line of work aimed at developing situation-based applications. However, as highlighted

in the previous section, there remain multiple open challenges needing a strong concerted community effort to really advance the field of multimodal situation recognition.

There have been multiple recent efforts in the community to spread awareness on the topic and also synergize the efforts aimed at handling different challenges identified. For example, a preliminary roadmap to connect personal data streams to aggregated (situational) multimodal streams was discussed in a workshop on “personal data meets distributed multimedia” at the 2013 ACM Multimedia Conference [Singh et al. 2013]. Similarly, there have been multiple recent tutorials on the topic of situation recognition organized at ACM Multimedia and ACM Multimedia Information Retrieval Conferences (e.g., Singh et al. [2016]).

While each of these marks a step toward concerted efforts in developing the field of multimedia situation recognition, there is a lot more that the field can gain from the wider expertise of multimedia researchers to create human-centric solutions. The multimedia research community is a melting pot of researchers with such varied expertise concept detection, information processing, social media analysis, system design, and human-computer interaction, and their collective wisdom is essential to making strong progress in the field of multimedia situation recognition.

The growth in this field could generate effective mechanisms to detect situations use multimodal data and then using mechanisms to persuade humans in the loop into taking the right actions at the right time. Ultimately, such an ability to build situation-aware human-in-the-loop systems could be pivotal in helping improve human lives in fields ranging from healthcare to traffic to urban planning and civic engagement.

Acknowledgments

This chapter summarizes recent work undertaken by the author in close collaboration with Ramesh Jain, Mingyan Gao, and Siripen Pongpaichet (who were also coauthors in the original papers).

Hence, while presenting a summary and vision in this chapter, the author would like to sincerely acknowledge the contributions of the above collaborators.

Hawkes Processes for Events in Social Media

Marian-Andrei Rizoiu (Australian National University and CSIRO Data61),

Young Lee (CSIRO Data61 and Australian National University),

Swapnil Mishra (Australian National University and CSIRO Data61),

Lexing Xie (Australian National University and CSIRO Data61)

This chapter provides an accessible introduction for point processes, and especially Hawkes processes, for modeling discrete, inter-dependent events over continuous time. We start by reviewing the definitions and key concepts in point processes. We then introduce the Hawkes process and its event intensity function, as well as schemes for event simulation and parameter estimation. We also describe a practical example drawn from social media data—we show how to model retweet cascades using a Hawkes self-exciting process. We present a design of the memory kernel, and results on estimating parameters and predicting popularity. The code and sample event data are available in an online repository.

8.1 Introduction

Point processes are collections of random points falling in some space, such as time and location. Point processes provide the statistical language to describe the timing and properties of events. Problems that fit this setting span a range of application domains. In finance, an event can represent a buy or a sell transaction on the stock market that influences future prices and volumes of such transactions. In geophysics, an event can be an earthquake that is indicative of the likelihood of

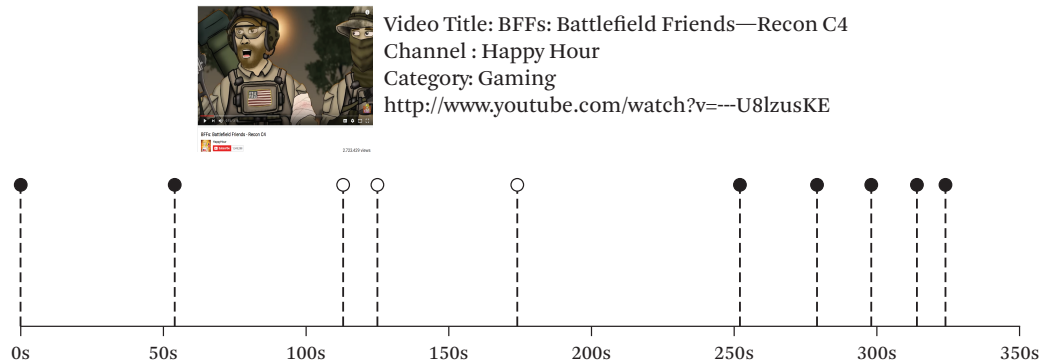


Figure 8.1 A point process, showing tweets about a gaming video (<http://www.youtube.com/watch?v=---U8lzusKE>) on YouTube. The first 10 events are shown. They correspond to the first 10 tweets in the diffusion, the time stamps of which are indicated by dashed vertical lines. An event with a hollow tip denotes a retweet of a previous tweet.

another earthquake in the vicinity in the immediate future. In ecology, event data consist of a set of point locations where a species has been observed. In the analysis of online social media, events can be user actions over time, each of which have a set of properties such as user influence, topic of interest, and connectivity of the surrounding network.

Figure 8.1 depicts an example point process: a retweet cascade about a Gaming YouTube video (YouTubeID `---U8lzusKE`). Here each tweet is an event that happens at a certain point in continuous time. Three of the events shown in Figure 8.1 are depicted using hollow tips—they are retweets of a previous tweet, or the act of one user re-sharing the content from another user. We explicitly observe information diffusion via retweets; however, there are other diffusion mechanisms that are not easily observed. These include offline *word-of-mouth* diffusion, or information propagating in emails and other online platforms. One way for modeling the overall information diffusion process is to use so-called *self-exciting processes*—in this type of processes the probability of seeing a new event increases due to previous events. Point-process models are useful for answering a range of different questions. These include *explaining* the nature of the underlying process, *simulating* future events, and *predicting* the likelihood and volume of future events.

In Section 8.2, we first review the basic concepts and properties of point processes in general, and of Poisson processes. These provide foundations for defining the Hawkes process. In Section 8.3, we introduce the Hawkes process—including expressions of the event rate and the underlying branching structure. Section 8.4 describes two procedures for sampling from a Hawkes process. One uses thinning, or rejection sampling, while the other makes use of a novel variable decomposi-

tion. Section 8.5 derives the likelihood of a Hawkes process and describes a maximum likelihood estimation procedure for its parameters, given observed event sequence(s). In the last section of this chapter we present an example of estimating Hawkes processes from a retweet event sequence. We introduce the data, the problem formulation, the fitting results, and interpretations of the model. We include code snippets for this example in Section 8.6.5; the accompanying software and data are included in an online repository (<https://github.com/s-mishra/featuredriven-hawkes>).

This chapter aims to provide a self-contained introduction to the fundamental concepts and methods for self-exciting point-processes, with a particular emphasis on the Hawkes process. The goal is for the readers to be able to understand the key mathematical and computational constructs of a point process, formulate their problems in the language of point processes, and use point process in domains including but not limited to modeling events in social media. The study of point processes has a long history, with discussions of the Hawkes process dating back at least to the early 1970s [Hawkes 1971]. Despite the richness of existing literature, we found through our own recent experience in learning and applying Hawkes processes that a self-contained tutorial centered around problem formulation and applications is still missing. This chapter aims at filling this gap, providing the foundations as well as an example of point processes for social media. Its intended audiences are aspiring researchers and beginning PhD students, as well as any technical readers with a special interest in point processes and their practical applications. For in-depth reading, we refer the readers to overview papers and books [Daley and Vere-Jones 2003, Toke 2011] on Hawkes processes. We note that this chapter does not cover other important variants used in the multimedia area, such as self-inhibiting processes [Yang et al. 2015], or non-causal processes (in time or space), such as the Markov point processes [Pham et al. 2016].

8.2 Preliminary: Poisson Processes

In this section, we introduce the fundamentals of point processes and their simplest subclass, the Poisson process. These serve as the foundation on which we build, in later sections, the more complex processes, such as the Hawkes point process.

8.2.1 Defining a Point Process

A point process on the nonnegative real line, where the nonnegative line is taken to represent time, is a random process whose realizations consist of the event times T_1, T_2, \dots of event times falling along the line. T_i can usually be interpreted as the time of occurrence of the i -th event, and T are often referred to as event times.

The equivalent counting process. A counting process N_t is a random function defined on time $t \geq 0$, and takes integer values $1, 2, \dots$. Its value is the number of events of the point process by time t . Therefore it is uniquely determined by a sequence of non-negative random variables T_i , satisfying $T_i < T_{i+1}$ if $T_i \leq \infty$. In other words, N_t counts the number of events up to time t , i.e.,

$$N_t := \sum_{i \geq 1} \mathbf{1}_{\{t \geq T_i\}}. \quad (8.1)$$

Here $\mathbf{1}_{\{\cdot\}}$ is the indicator function that takes value 1 when the condition is true, and 0 otherwise. We can see that $N_0 = 0$. N_t is piecewise constant and has jump size of 1 at the event times T_i . It is easy to see that the set of event times T_1, T_2, \dots and the corresponding counting process are equivalent representations of the underlying point process.

8.2.2 Poisson Processes: Definition

The simplest class of point process is the Poisson process.

Definition 8.1 (Poisson process.) Let $(\tau_i)_{i \geq 1}$ be a sequence of *i.i.d.* exponential random variables with parameter λ and event times $T_n = \sum_{i=1}^n \tau_i$. The process $(N_t, t \geq 0)$ defined by $N_t := \sum_{i \geq 1} \mathbf{1}_{\{t \geq T_i\}}$ is called a *Poisson process* with intensity λ .

Event intensity λ . The sequence of τ_j are called the *inter-arrival times*, i.e., the first event occurs at time τ_1 , the second occurs at τ_2 after the first, etc. The inter-arrival times τ_i are independent, and each of them follows an exponential distribution with parameter λ . Here, the notation $f_\tau(t)$ denotes the probability density function of random variable τ taking values denoted by t .

$$f_\tau(t) = \begin{cases} \lambda e^{-\lambda t}, & \text{if } t \geq 0 \\ 0, & \text{if } t < 0. \end{cases} \quad (8.2)$$

Here $\lambda > 0$ is a positive constant. The expected value of τ_i can be computed in closed form, as follows:

$$\begin{aligned} \mathbb{E}_\tau[\tau] &= \int_0^\infty t f_\tau(t) dt = \lambda \int_0^\infty t e^{-\lambda t} dt = \left[-t e^{-\lambda t} \right]_{t=0}^{t=\infty} + \int_0^\infty e^{-\lambda t} dt \\ &= 0 - \left[\frac{1}{\lambda} e^{-\lambda t} \right]_{t=0}^{t=\infty} = \frac{1}{\lambda}. \end{aligned} \quad (8.3)$$

Intuitively, events are arriving at an average rate of λ per unit time, since the expected time between event times is λ^{-1} . Hence we say, informally, that the Poisson

process has *intensity* λ . In general, the event intensity needs not be constant, but is a function of time, written as $\lambda(t)$. This general case is called a *non-homogeneous Poisson process*, and will be discussed in Section 8.2.4.

Arrival times and counting process. The *arrival times*, or the event times, are given by:

$$T_n = \sum_{j=1}^n \tau_j, \quad (8.4)$$

where T_n is the time of the n -th arrival. The event times T_1, T_2, \dots form a random configuration of points on the real line $[0, \infty)$ and N_t counts the number of such ones in the interval $[0, t]$. Consequently, N_t increments by one for each T_i . This can be explicitly written as follows.

$$N_t = \begin{cases} 0, & \text{if } 0 \leq t < T_1 \\ 1, & \text{if } T_1 \leq t < T_2 \\ 2, & \text{if } T_2 \leq t < T_3 \\ \vdots & \\ n, & \text{if } T_n \leq t < T_{n+1}, \\ \vdots & \end{cases} \quad (8.5)$$

We observe that N_t is defined so that it is *right continuous with left limits*. The left limit $N_{t-} = \lim_{s \uparrow t} N_s$ exists and $N_{t+} = \lim_{s \downarrow t} N_s$ exists and is taken to be N_t .

8.2.3 The Memorylessness Property of Poisson Processes

Being *memoryless* in a point process means that the distribution of future inter-arrival times depends only on relevant information about the current time, but not on information from further in the past. We show that this is the case for Poisson processes.

We compute the probability of observing an inter-arrival time τ longer than a predefined time length t . F_τ is the cumulative distribution function of the random variable τ , which is defined as $F_\tau(t) := \mathbb{P}\{\tau \leq t\}$. We have

$$F_\tau(t) := \mathbb{P}(\tau \leq t) = \int_0^t \lambda e^{-\lambda x} dx = \left[-e^{-\lambda x} \right]_{x=0}^{x=t} = 1 - e^{-\lambda t}, \quad t \geq 0, \quad (8.6)$$

and hence the probability of observing an event at time $\tau > t$ is given by

$$\mathbb{P}(\tau > t) = e^{-\lambda t}, \quad t \geq 0. \quad (8.7)$$

Suppose we were waiting for an arrival of an event, say a tweet, the inter-arrival times of which follow an exponential distribution with parameter λ . Assume that m time units have elapsed and during this period no events have arrived, i.e., there are no events during the time interval $[0, m]$. The probability that we will have to wait a further t time units is given by

$$\begin{aligned}\mathbb{P}(\tau > t + m | \tau > m) &= \frac{\mathbb{P}(\tau > t + m, \tau > m)}{\mathbb{P}(\tau > m)} \\ &= \frac{\mathbb{P}(\tau > t + m)}{\mathbb{P}(\tau > m)} = \frac{e^{-\lambda(t+m)}}{e^{-\lambda m}} = e^{-\lambda t} = \mathbb{P}(\tau > t).\end{aligned}\quad (8.8)$$

In this derivation, we first expand the conditional probability using Bayes' rule. The next step follows from the fact that $\tau > m$ always holds when $\tau > t + m$. The last step follows from Equation (8.7).

Equation (8.8) denotes the *memorylessness* property of Poisson processes. That is, the probability of having to wait an additional t time units after already having waited m time units is the same as the probability of having to wait t time units when starting at time 0. Putting it differently, if one interprets τ as the time of arrival of an event where τ follows an exponential distribution, the distribution of $\tau - m$ given $\tau > m$ is the same as the distribution of τ itself.

8.2.4 Non-homogeneous Poisson Processes

In Poisson processes, events arrive randomly with the constant intensity λ . This initial model is sufficient for describing simple processes, say the arrival of cars on a street over a short period of time. However, we need to be able to vary the event intensity with time in order to describe more complex processes, such as simulating the arrivals of cars during rush hours and off-peak times. In a non-homogeneous Poisson process, the rate of event arrivals is a function of time, i.e., $\lambda = \lambda(t)$.

Definition 8.2 A point process $\{N_t\}_{t \geq 0}$ can be completely characterized by its conditional intensity function, defined as

$$\lambda(t | \mathcal{H}_t) = \lim_{h \rightarrow 0} \frac{\mathbb{P}\{N_{t+h} - N_t = 1 | \mathcal{H}_t\}}{h} \quad (8.9)$$

where \mathcal{H}_t is the history of the process up to time t , containing the list of event times $\{T_1, T_2, \dots, T_{N_t}\}$.

In the rest of this chapter, we use the shorthand notation $\lambda(t) =: \lambda(t | \mathcal{H}_t)$, always assuming an implicit history before time t . The above definition gives the intensity view of a point process, equivalent with the two previously defined views with event times and the counting process. In other words, the event intensity $\lambda(t)$ determines

the distribution of event times, which in turn determine the counting process. Formally, $\lambda(t)$ and N_t are related through the probability of an event in a small time interval h :

$$\begin{aligned}\mathbb{P}(N_{t+h} = n + m \mid N_t = n) &= \lambda(t)h + o(h) && \text{if } m = 1 \\ \mathbb{P}(N_{t+h} = n + m \mid N_t = n) &= o(h) && \text{if } m > 1 \\ \mathbb{P}(N_{t+h} = n + m \mid N_t = n) &= 1 - \lambda(t)h + o(h) && \text{if } m = 0,\end{aligned}\tag{8.10}$$

where $o(h)$ is a function so that $\lim_{h \downarrow 0} \frac{o(h)}{h} = 0$. In other words, the probability of observing an event during the infinitesimal interval of time t and $t + h$ when $h \downarrow 0$ is $\lambda(t)h$. The probability of observing more than one event during the same interval is negligible.

8.3 Hawkes Processes

In the models described in the previous section, the events arrive independently, either at a constant rate (for the Poisson process) or governed by an intensity function (for the non-homogeneous Poisson). However, for some applications, it is known that the arrival of an event increases the likelihood of observing events in the near future. This is the case of earthquake aftershocks when modeling seismicity, or that of user interactions when modeling preferential attachment in social networks. In this section, we introduce a class of processes in which the event arrival rate explicitly depends on past events—i.e., *self-exciting processes*—and we further detail the most well-known self-exciting process, the Hawkes process.

8.3.1 Self-exciting Processes

A self-exciting process is a point process in which the arrival of an event causes the conditional intensity function to increase. A well-known self-exciting process was proposed by Hawkes [1971], and it is based on a counting process in which the intensity function depends explicitly on all previously occurring events. The Hawkes process is defined as follows:

Definition 8.3 (Hawkes process) Let $\{N_t\}_{t \geq 0}$ be a counting process with associated history \mathcal{H}_t , $t \geq 0$. The point process is defined by the event intensity function $\lambda(t)$ with respect to Equation (8.10) (the intensity view of a non-homogeneous Poisson process). The point process is said to be a Hawkes process if the conditional intensity function $\lambda(t|\mathcal{H}_t)$ takes the form:

$$\lambda(t|\mathcal{H}_t) = \lambda_0(t) + \sum_{i:t > T_i} \phi(t - T_i),\tag{8.11}$$

where $T_i < t$ are all the event times having occurred before current time t , and which contribute to the event intensity at time t . $\lambda_0(t) : \mathbb{R} \mapsto \mathbb{R}_+$ is a deterministic base intensity function, and $\phi : \mathbb{R} \mapsto \mathbb{R}_+$ is called the memory kernel—both of which are further detailed in the next section. We observe that the Hawkes process is a particular case of a non-homogeneous Poisson process, in which the intensity is stochastic and explicitly depends on previous events through the kernel function $\phi(\cdot)$.

8.3.2 The Intensity Function

The quantity $\lambda_0(t) > 0$ is the base (or background) intensity, describing the arrival of events triggered by external sources. These events are also known as *exogenous* or *immigrant* events, and their arrival is independent of the previous events within the process. The self-exciting flavor of the Hawkes process arises through the summation term in Equation (8.11), where the kernel $\phi(t - T_i)$ modulates the change that an event at time T_i has on the intensity function at time t . Typically, the function $\phi(\cdot)$ is taken to be monotonically decreasing so that more recent events have higher influence on the current event intensity, compared to events having occurred further away in time. Figure 8.2(a) shows an example realization of a Hawkes process: nine events are observed, at times T_1, T_2, \dots, T_9 , and their corresponding inter-arrival times $\tau_1, \tau_2, \dots, \tau_9$. Fig 8.2(b) shows the corresponding counting process N_t over time, which increases by one unit for each T_i as defined in Equation (8.5). Figure 8.2(c) shows the intensity function $\lambda(t)$ over time. Visibly, the value of the intensity function increases suddenly, immediately at the occurrence of an event T_i , and diminishes as time passes and the effect of the given event T_i decays.

Choice of the kernel ϕ . The kernel function $\phi(\cdot)$ does not have to be monotonically decreasing. However, in this chapter we restrict the discussion to the decreasing families of functions, given that it is natural to see the influence of an event decay over time, as shown in Section 8.6. A popular decay function is the exponential function [Hawkes 1971], taking the following form:

$$\phi(x) = \alpha e^{-\delta x}, \quad (8.12)$$

where $\alpha \geq 0$, $\delta > 0$, and $\alpha < \delta$. Another kernel that is widely used in the literature is the power-law kernel:

$$\phi(x) = \frac{\alpha}{(x + \delta)^{\eta+1}}, \quad (8.13)$$

where $\alpha \geq 0$, $\delta, \eta > 0$, and $\alpha < \eta\delta^\eta$. This kernel is commonly used within the seismology literature [Ozaki 1979] and in the social media literature [RizoIU et al. 2017].

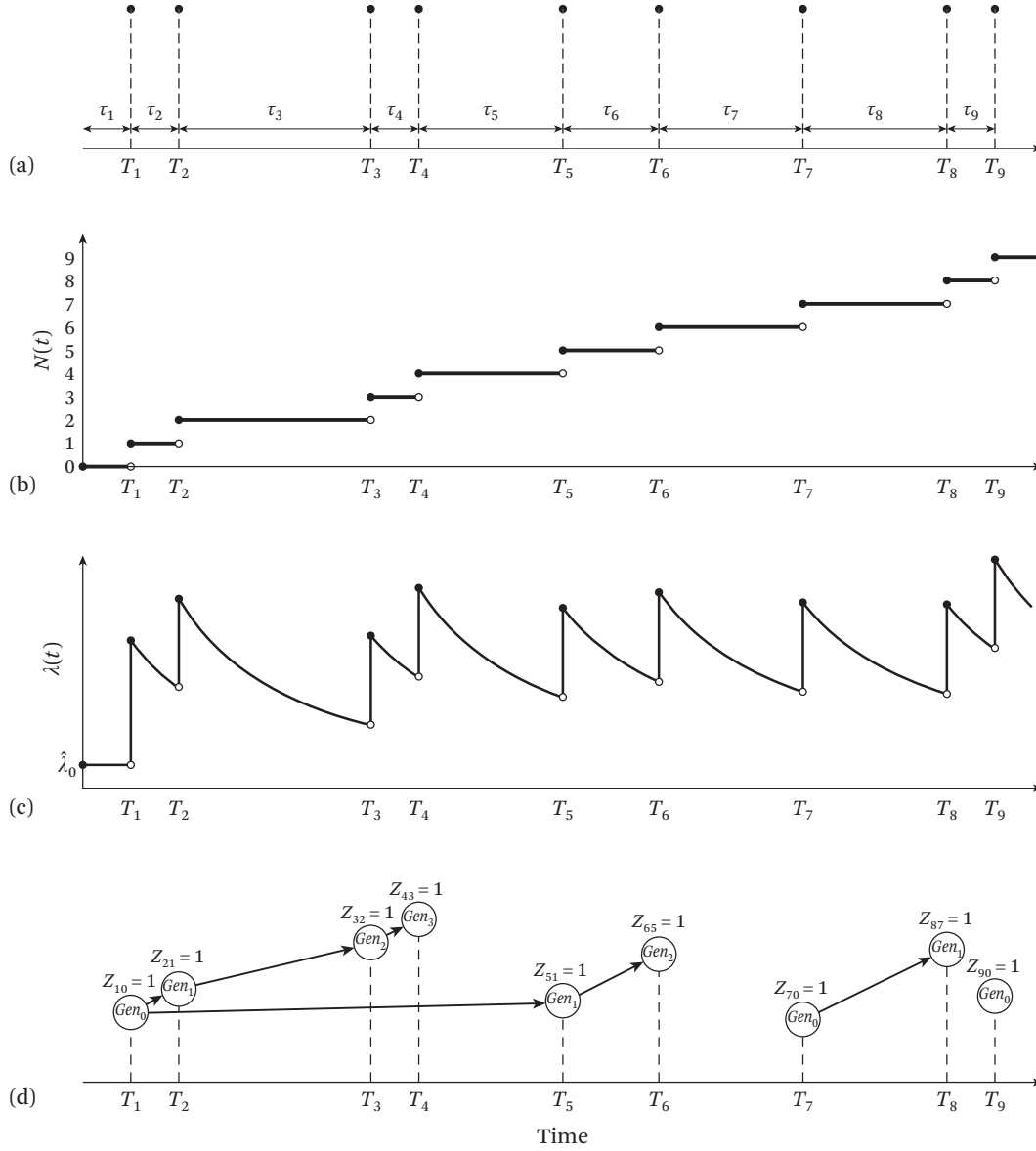


Figure 8.2 Hawkes process with an exponential decay kernel. (a) The first nine event times are shown. T_i represent event times, while τ_i represent inter-arrival times. (b) Counting process over time. N_i increases by one unit at each event time T_i . (c) Intensity function over time. Note how each event provokes a jump, followed by an exponential decay. Later decays unfold on top of the tail of earlier decays, resulting in apparently different decay rates. (d) The latent or unobserved branching structure of the Hawkes process. Every circle represents one event having occurred at T_i ; the arrows represent the root-offspring relation. Gen_i specifies the generation of the event, with $i = 0$ for immigrants or $i > 0$ for the offspring. Z_{ij} are random variables, such that $Z_{i0} = 1$ if event i is an immigrant, and $Z_{ij} = 1$ if event i is an offspring of event j .

The exponential kernel defined by Equation 8.12 is typically the popular choice of kernel with Hawkes processes [Embrechts et al. 2011], unless demanded otherwise by the phenomena modeled using the self-exciting process (for example, we use a power-law kernel for modeling information diffusion in social media, in Section 8.6).

Other self-exciting point processes. These have been proposed, which follow the canonical specification given in Equation (8.11) and which extend the initial self-exciting process proposed by Hawkes [1971]. We do not cover these processes in this chapter; however, we advise the reader of Hawkes extensions such as the non-linear Hawkes processes [Brémaud and Massoulié 1996, Daley and Vere-Jones 2003], the general space time self-exciting point process [Veen and Schoenberg 2008, Ogata 1988], processes with exponential base event intensity [Dassios and Zhao 2011], or self-inhibiting processes [Yang et al. 2015].

8.3.3 The Branching Structure

Another equivalent view of the Hawkes process refers to the Poisson cluster process interpretation [Hawkes and Oakes 1974], which separates the events in a Hawkes process into two categories: *immigrants* and *offspring*. The offspring events are triggered by existing (previous) events in the process, while the immigrants arrive independently and thus do not have an existing parent event. The offspring are said to be structured into *clusters*, associated with each immigrant event. This is called the *branching structure*. In the rest of this section, we further detail the branching structure and we compute two quantities: the *branching factor*—the expected number of events directly triggered by a given event in a Hawkes process—and the estimated total number of events in a cluster of offspring. As shown in Section 8.6, both of these quantities become very important when the Hawkes processes are applied to practical domains, such as online social media.

An example branching structure. We consider the case immigrant events follow a homogeneous Poisson process with base intensity $\lambda_0(t)$, while offspring are generated through the self-excitement, governed by the summation term in Equation (8.32). Figure 8.2(d) illustrates the branching structure of the nine event times of the example Hawkes process discussed earlier. Event times T_i are denoted by circles and the “parent-offspring” relations between the events are shown by arrows. We introduce the random variables Z_{ij} , where $Z_{i0} = 1$ if event i is an immigrant, and $Z_{ij} = 1$ if event i is an offspring of event j . The text in each circle denotes the generation to which the event belongs, i.e., Gen_k denotes the k -th generation. Immigrants

are labeled as $\mathcal{G}en_0$, while generations $\mathcal{G}en_k$, $k > 0$ denote their offspring. For example, T_3 and T_6 are immediate offspring of the immigrant T_2 , i.e., mathematically expressible as $Z_{32} = 1$, $Z_{62} = 1$, and $Z_{20} = 1$.

The cluster representation states that the immediate offspring events associated with a particular parent arrive according to a non-homogeneous Poisson process with intensity $\phi(\cdot)$, i.e., T_3 and T_6 are event realizations coming from a non-homogeneous Poisson process endowed with the intensity $\phi(t - T_2)$ for $t > T_2$. The event that produces an offspring is described as the immediate ancestor or root of the offspring; T_7 is the immediate ancestor of T_8 . The events that are directly or indirectly connected to an immigrant form the *cluster* of offspring associated with that immigrant; e.g., T_1 is an immigrant and T_2, T_3, T_4, T_5 , and T_6 form its cluster of offspring. Similarly, T_7 and T_8 form another cluster. Finally, T_9 is a cluster by itself.

Branching factor (branching ratio). One key quantity that describes the Hawkes process is its branching factor n^* , defined as the expected number of direct offspring spawned by a single event. The branching factor n^* intuitively describes the amount of events to appear in the process, or informally, *virality* in the social media context. In addition, the branching factor gives an indication about whether the cluster of offspring associated with an immigrant is an infinite set. For $n^* < 1$, the process is in a *subcritical regime*: the total number of events in any cluster is bounded. Immigrant events occur according to the base intensity $\lambda_0(t)$, but each one of them has associated with it a finite cluster of offspring, both in number and time extent. When $n^* > 1$, the process is in a so-called *supercritical regime* with $\lambda(t)$ increasing and the total number of events in each cluster being unbounded. We compute the branching factor by integrating $\phi(t)$ —the contribution of each event—over event time t :

$$n^* = \int_0^{\infty} \phi(\tau) d\tau. \quad (8.14)$$

Expected number of events in a cluster of offspring. The branching factor n^* indicates whether the number of offspring associated with each immigrant is finite ($n^* < 1$) or infinite ($n^* > 1$). When $n^* < 1$ a more accurate estimate of the size of each cluster can be obtained. Let A_i be the expected number of events in *Generation* _{i} , and $A_0 = 1$ (as each cluster has only one immigrant). The expected number of total events in the cluster, N_{∞} , is defined as:

$$N_{\infty} = \sum_{i=0}^{\infty} A_i. \quad (8.15)$$

To compute $A_i, i \geq 1$, we notice that each of the A_{i-1} events in the previous generation has on average n^* children events. This leads to an inductive relationship $A_i = A_{i-1}n^*$. Knowing that $A_0 = 1$, we derive:

$$A_i = A_{i-1}n^* = A_{i-2}(n^*)^2 = \dots = A_0(n^*)^i = (n^*)^i, i \geq 1. \quad (8.16)$$

We obtain an estimate of the size of each cluster of immigrants N_∞ as the sum of a converging geometric progression (assuming $n^* < 1$):

$$N_\infty = \sum_{i=0}^{\infty} A_i = \frac{1}{1 - n^*} \quad \text{where } n^* < 1. \quad (8.17)$$

8.4 Simulating Events from Hawkes Processes

In this section, we focus on the problem of simulating a series of random events according to the specifications of a given Hawkes process. This is useful for gathering statistics about the process, and can form the basis for diagnostics, inference, or parameter estimation. We present two simulation techniques for Hawkes processes. The first technique, the thinning algorithm [Ogata 1981], applies to all non-homogeneous Poisson processes, and can be applied to Hawkes processes with any kernel function $\phi(\cdot)$. The second technique, recently proposed by Dassios and Zhao [2013], is computationally more efficient, as it designs a variable decomposition technique for Hawkes processes with exponential decaying kernels.

8.4.1 The Thinning Algorithm

The basic goal of a sampling algorithm is to simulate inter-arrival times $\tau_i, i = 1, 2, \dots$ according to an intensity function λ_i . We first review the sampling method for a homogeneous Poisson process, then we introduce the thinning (or additive) property of Poisson processes, and we use this to derive the sampling algorithm for Hawkes processes.

Inter-arrival times in a homogeneous Poisson process follow an exponential distribution as specified in 8.3: $f_\tau(t) = \lambda e^{-\lambda t}, t > 0$ and its cumulative distribution function is $F_\tau(t) = 1 - e^{-\lambda t}$. Because both $F_\tau(t)$ and $F_\tau^{-1}(t)$ have a closed-form expression, we can use the *inverse transform sampling* technique to sample waiting times. Intuitively, if X is a random variable with the cumulative distribution function F_X and $Y = F_X(X)$ is a uniformly distributed random variable ($\sim U(0, 1)$), then $X^* = F_X^{-1}(Y)$ has the same distribution as X . In other words, sampling $X^* = F_X^{-1}(Y), Y \sim U(0, 1)$ is identical with sampling X . For the exponentially distributed waiting times of the Poisson process, the inverse cumulative distribution function

has the form $F_\tau^{-1}(u) = \frac{-\ln u}{\lambda}$. Consequently, sampling a waiting interval τ in a Poisson process is simply:

$$\text{Sample } u \sim U(0, 1), \text{ then compute } \tau = \frac{-\ln u}{\lambda} \quad (8.18)$$

The thinning property of the Poisson processes states that a Poisson process with intensity λ can be split into two independent processes with intensities λ_1 and λ_2 , so that $\lambda = \lambda_1 + \lambda_2$. In other words, each event of the original process can be assigned to one of the two new processes that are running independently. From this property, we can see that we can simulate a non-homogeneous Poisson process with the intensity function $\lambda(t)$ by *thinning* a homogeneous Poisson process with the intensity $\lambda^* \geq \lambda(t)$, $\forall t$.

A thinning algorithm to simulate Hawkes processes is presented in Algorithm 8.1. For any bounded $\lambda(t)$, we can find a constant λ^* so that $\lambda(t) \leq \lambda^*$ in a given time interval. In particular, for Hawkes processes with a monotonically decreasing kernel function $\phi(t)$, it is easy to see that between two consecutive event times $[T_i, T_{i+1})$, $\lambda(T_i)$ is the upper bound of event intensity. We exemplify the sampling of event time T_{i+1} , after having already sampled T_1, T_2, \dots, T_i . We start our time counter $T = T_i$. We sample an inter-arrival time τ , using Equation (8.18), with $\lambda^* = \lambda(T)$ and we update the time counter $T = T + \tau$ (steps 3(a) to 3(c) in Algorithm 8.1). We accept or reject this inter-arrival time according to the ratio of the true event rate to the thinning rate λ^* (step 3(e)). If accepted, we record the event time $i + 1$ as $T_{i+1} = T$. Otherwise, we repeat the sampling of an inter-arrival time until one is accepted. Note that, even if an inter-arrival time is rejected, the time counter T is still updated, i.e., the principle of thinning a homogeneous Poisson

Algorithm 8.1 Simulation of a Hawkes process by thinning

1. Given Hawkes process as in Eq (8.11)
2. Set current time $T = 0$ and event counter $i = 1$
3. **while** $i \leq N$.
 - (a) Set the upper bound of Poisson intensity $\lambda^* = \lambda(T)$ (using Eq (8.11))
 - (b) Sample inter-arrival time: draw $u \sim U(0, 1)$ and let $\tau = -\frac{\ln(u)}{\lambda^*}$
(as described in Eq (8.18))
 - (c) Update current time: $T = T + \tau$
 - (d) Draw $s \sim U(0, 1)$
 - (e) **if** $s \leq \frac{\lambda(T)}{\lambda^*}$, accept the current sample: let $T_i = T$ and $i = i + 1$
otherwise reject the sample, and return to step (a)

process with a higher intensity value. Also note that, for efficiency reasons, the upper bound λ^* can be updated even in the case of a rejected inter-arrival time, given the strict monotonicity of $\lambda(t)$ in-between event times. The temporal complexity of sampling N events is $O(N^2)$, since brute-force computation of event intensity using Eq (8.11) is $O(N)$. Furthermore, if event rates decay fast, then the number of rejected samples can be high before there is an accepted new event time.

8.4.2 Efficient Sampling by Decomposition

We now outline a more efficient sampling algorithm for Hawkes processes with an exponential kernel that does not resort to rejection sampling. Recently proposed by [Dassios and Zhao \[2013\]](#), it scales linearly to the number of events drawn.

First, the proposed algorithm applies to a Hawkes process with exponential immigrant rates and exponential memory kernel. This is a more general form than what we defined in Section 8.3.2. The immigrant rate is described by a non-homogeneous Poisson process following an exponential function $a + (\lambda_0 - a)e^{-\delta t}$. For each new event, the *jump* it introduces in event intensity is described by a constant γ :

$$\lambda(t) = a + (\lambda_0 - a)e^{-\delta t} + \sum_{T_i < t} \gamma e^{-\delta(t-T_i)}, \quad t > 0. \quad (8.19)$$

We can envision generalizing this even more by introducing a distribution to *gamma*, but this is out of scope for this tutorial.

We note that a process is a Markov process if it has the property that, conditional on the present, the future is independent of the past. [Ogata \[1981\]](#) has shown that the intensity process is a Markov process when ϕ is exponential. This can be intuitively understood for the event intensity function above, due to $\lambda(t_2) = e^{-\delta(t_2-t_1)}\lambda(t_1)$, for any $t_2 > t_1$. In other words, given current event intensity $\lambda(t_1)$, future intensity only depends on the time elapsed since time t_1 .

We use this Markov property to decompose the inter-arrival times into two independent simpler random variables. The first random variable s_0 represents the inter-arrival time of the next event, if it were to come from the constant background rate a . It is easy to see that this is sampled according to Eq (8.18). The second random variable s_1 represents the inter-arrival time of the next event if it were to come from either the exponential immigrant kernel $(\lambda_0 - a)e^{-\delta t}$ or the Hawkes self-exciting kernels from each of the past events $\sum_{T_i < t} e^{-\delta(t-T_i)}$. The cumulative distribution function of s_1 can be explicitly inverted due to its Markov property: a full derivation can be found in [Dassios and Zhao \[2013\]](#). Intuitively, the sampled inter-arrival time is the minimum of these two cases. It is also worth noting that the second arrival time may not be finite: this is expected, as the exponential kernel

Algorithm 8.2 Efficient simulation of a Hawkes process with exponential kernel

1. Set $T_0 = 0$, initial event rate $\lambda(T_0) = \lambda_0$
2. **for** $i = 1, 2, \dots, N$
 - (a) Draw $u_0 \sim U(0, 1)$ and set $s_0 = -\frac{1}{a} \ln u_0$
 - (b) Draw $u_1 \sim U(0, 1)$; Set $d = 1 + \frac{\delta \ln u_1}{\lambda(T_{i-1}^+) - a}$
 - (c) **if** $d > 0$, set $s_1 = -\frac{1}{\delta} \ln d$, $\tau_i = \min\{s_0, s_1\}$
otherwise $\tau_i = s_0$
 - (d) Record the i^{th} jump time $T_i = T_{i-1} + \tau_i$
 - (e) Update event intensity at the left side of T_i with exponential decay
 $\lambda(T_i^-) = (\lambda(T_{i-1}^+) - a)e^{-\delta\tau_i} + a$
 - (f) Update event intensity at the right side of T_i with a jump from the i^{th} event
 $\lambda(T_i^+) = \lambda(T_i^-) + \gamma$

decays fast. In this case, the next event will be an immigrant from the constant rate. This is outlined in Algorithm 8.2.

This algorithm is efficient because the intensity function can be updated in constant time for each event with steps 2(e) and 2(f), and because this algorithm does not rely on rejection sampling. The decomposition method above cannot be easily used on the power law kernel, since the power law does not have the Markov property.

8.5 Estimation of Hawkes Processes Parameters

One challenge when modeling using self-exciting point processes is estimating parameters from observed data. In the case of the Hawkes process with exponential kernel, one would typically have to determine the function $\lambda_0(t)$ (the base intensity defined in Equation 8.11), and the values of the parameters of the decaying kernel $\phi(t)$ (α and δ ; see Equation 8.19). One can achieve this by maximizing the likelihood over the observed data. In Section 8.5.1 we derive the formula of the likelihood function for a Hawkes process and in Section 8.5.2 we discuss a few practical concerns of using maximum likelihood estimation.

8.5.1 Likelihood Function for Hawkes Process

Let $N(t)$ be a point process on $[0, T]$ for $T < \infty$ and let $\{T_1, T_2, \dots, T_n\}$ denote a realization, i.e., the set of event times, of $N(t)$ over the period $[0, T]$. Then the data likelihood L as a function of parameter set θ is:

$$L(\theta) = \prod_{i=1}^n \lambda(T_i) e^{-\int_0^T \lambda(t) dt}. \quad (8.20)$$

We sketch the derivation of the likelihood formula, along the lines of Daley and Vere-Jones [2003], Laub et al. [2015], and Rasmussen [2013]. If we are currently at some time t , recall that the history \mathcal{H}_t is the list of times of events T_1, T_2, \dots, T_n up to but not including time t . Borrowing the $*$ notation from Daley and Vere-Jones [2003], we define $f^*(t) := f(t|\mathcal{H}_t)$ to be the conditional probability density function of the time of the next event T_{n+1} given the history of previous event T_1, T_2, \dots, T_n . Recall that $\mathbb{P}\{T_{n+1} \in (t, t + dt)\} = f_{T_{n+1}}(t)dt$. We have

$$f(T_1, T_2, \dots, T_n) = \prod_{i=1}^n f(T_i|T_1, T_2, \dots, T_{i-1}) = \prod_{i=1}^n f^*(T_i). \quad (8.21)$$

It turns out that the event intensity $\lambda(t)$ can be expressed in terms of the conditional density f^* and its corresponding cumulative distribution function F^* [Rasmussen 2011]:

$$\lambda(t) = \frac{f^*(t)}{1 - F^*(t)}. \quad (8.22)$$

The expression above is given without a formal proof, but it can be interpreted heuristically as follows. Consider an infinitesimal interval dt around t , with $f^*(t)dt$ corresponding to the probability that there is an event in dt , and $1 - F^*(t)$ corresponding to the probability of no new events before time t . After manipulating the expression using Bayes' rule [Rasmussen 2011], the ratio of the two can be shown to be equivalent to the expectation of an increment of the counting process $N_{t+dt} - N_t$, which by Eq (8.10) is essentially $\lambda(t)dt$.

We can write the conditional intensity function in terms of the cumulative distribution function F^* :

$$\lambda(t) = \frac{f^*(t)}{1 - F^*(t)} = \frac{\frac{\partial}{\partial t} F^*(t)}{1 - F^*(t)} = -\frac{\partial}{\partial t} \log(1 - F^*(t)). \quad (8.23)$$

Denote the last known event time before t as T_n ; then integrating both sides from (T_n, t) , we get

$$\int_{T_n}^t \lambda(s)ds = -[\log(1 - F^*(t)) - \log(1 - F^*(T_n))]. \quad (8.24)$$

Note that $F^*(T_n) = 0$ since $T_{n+1} > T_n$, and so

$$\int_{T_n}^t \lambda(s)ds = -\log(1 - F^*(t)). \quad (8.25)$$

Rearranging gives the following expression:

$$F^*(t) = 1 - \exp\left(-\int_{T_n}^t \lambda(s)ds\right). \quad (8.26)$$

Combining the relationship between $\lambda(t)$, $f^*(t)$, and $F^*(t)$ in Eq 8.22 gives

$$f^*(t) = \lambda(t) (1 - F^*(t)) = \lambda(t) \exp\left(-\int_{T_n}^t \lambda(s)ds\right). \quad (8.27)$$

Plugging Eq (8.27) above into the likelihood function, and combining integration ranges, we get the likelihood expression

$$L(\theta) = \prod_{i=1}^n f^*(T_i) = \prod_{i=1}^n \lambda(T_i) e^{-\int_{T_{i-1}}^{T_i} \lambda(u)du} = \prod_{i=1}^n \lambda(T_i) e^{-\int_0^{T_n} \lambda(u)du}. \quad (8.28)$$

8.5.2 Maximum Likelihood Estimation

Let θ be the set of parameters of the Hawkes process; its maximum likelihood estimate can be found by maximizing the likelihood function in Equation 8.20 with respect to θ over the space of parameter Θ . More precisely, the maximum likelihood estimate $\hat{\theta}$ is defined to be $\hat{\theta} = \arg \max_{\theta \in \Theta} l(\theta)$. From a standpoint of computational and numerical complexity, we note that summing is less expensive than multiplication. But more importantly, likelihoods would become very small and would risk running out of floating point precision very quickly, yielding an underflow; thus it is customary to maximize the log of the likelihood function:

$$l(\theta) = \log L(\theta) = -\int_0^T \lambda(t)dt + \sum_{i=1}^{N(T)} \log \lambda(T_i). \quad (8.29)$$

The natural logarithm is a monotonic function and maximizing the log-likelihood automatically implies maximizing the likelihood function. The negative log-likelihood can be minimized with optimization packages for non-linear objectives, such as the L-BFGS [Zhu et al. 1997] software.

Local maxima. One may run into problems of multiple local maxima in the log-likelihood. The shape of the negative log-likelihood function can be fairly complex and may not be globally convex. Due to the possible non-convex nature of the log-likelihood, performing maximum likelihood estimation would result in the estimate being the local maximum rather than the global maximum. A usual approach

used in trying to identify the global maximum involves using several sets of different initial values for the maximum likelihood estimation. Note that this does not mitigate the problem entirely, and it is quite possible that a local maximum may still be wrongly established as the global maximum. Alternatively, one can use different optimization methods in conjunction with several different sets of initial values. If the differing optimizations result in a consistent set of calibrated parameters, then we can have a higher certainty that the calibrated point is the actual global maximum.

Edge effects. Recall that N_t is the number of “arrivals” or “events” of the process by time t and that the sequence of event times T_1, T_2, \dots, T_{N_T} is assumed to be observed within the time interval $[0, T]$, where $T < \infty$. As discussed in Section 8.3.3, in a Hawkes process, the events usually arrive clustered in time: an immigrant and its offspring. In practical applications, the process might have started sometime in the past, prior to the moment when we start observing it, denoted as $t = 0$. Hence, there may be unobserved event times which occurred before time 0, which could have generated offspring events during the interval $[0, T]$. It is possible that the unobserved event times have an impact during the observation period, i.e., sometime after $t > 0$, but because we are not aware of them, their contribution to the event intensity is not recorded. Such phenomena are referred to as *edge effects* and are discussed in Daley and Vere-Jones [2003] and Rasmussen [2013]. One possible avenue to address this issue is to assume that the initial value of the intensity process equals the base intensity and disregard edge effects from event times occurring before the observation period; see Daley and Vere-Jones [2003]. This is usually the modeling setup in most applications within the Hawkes literature. As pointed out by Rasmussen [2013], the edge effects on the estimated model would turn out to be negligible if the dataset used is large enough. In this chapter, we set the base intensity to be a constant $\lambda(0) = \lambda_0$ and ignore edge effects from events that have occurred before the start of the observation period. For detailed discussions on handling edge effects, we refer the reader to the extensive work of Baddeley and Turner [2000], Bebbington and Harte [2001], Daley and Vere-Jones [2003], Møller and Rasmussen [2005], and Rasmussen [2013], which is summarized in Lapham [2014].

Computational bottleneck. A major issue with maximum likelihood estimation for Hawkes is the computational costs for evaluating the log-likelihood, in particular the evaluation of the intensity function, as shown hereafter. Note that the two components of the log-likelihood in Equation (8.29) can be maximized separately

if they do not have common terms; see [Ogata \[1988\]](#), [Daley and Vere-Jones \[2003\]](#), and [Zipkin et al. \[2016\]](#). The computational complexity arises due to the calculation of a double summation operation. This double sum comes from the second part of the log-likelihood:

$$\sum_{i=1}^{N_T} \log \lambda(T_i) = \sum_{i=1}^{N_T} \left(\log \left(a + (\lambda_0 - a)e^{-\delta t} + \sum_{j:T_j < T_i} \alpha e^{-\delta(T_i - T_j)} \right) \right). \quad (8.30)$$

Note that the complexity for most Hawkes processes is usually of the order $\mathcal{O}(N_T^2)$, where N_T is the number of event times. Hence estimating the parameters can be relatively slow when N_T is a big number, and it may be exacerbated if loop calculations cannot be avoided. In the case of an exponential kernel function, the number of operations required to evaluate Equation (8.30) can be reduced to $\mathcal{O}(N_T)$ using a recursive formula [[Ogata 1981](#)]. For a more complicated Hawkes process involving a power-law decay kernel, such as the epidemic type aftershock-sequences (ETAS) model [[Ogata 1988](#)] or the social media kernel constructed in Section 8.6, this strategy does not hold. For the ETAS model, the event intensity is defined as:

$$\lambda(t) = \lambda_0 + \sum_{i:t > T_i} \alpha \frac{e^{\delta \eta_1}}{(t - T_i + \gamma)^{\eta_2 + 1}} \quad (8.31)$$

for some constants $\lambda_0, \alpha, \eta_1, \gamma, \eta_2$. The ETAS model is a point process typically used to represent the temporal activity of earthquakes for a certain geophysical region. To reduce the computational complexity for the ETAS model, [Ogata et al. \[1993\]](#) presented a methodology that involved multiple transformations and numerical integration. They showed that there is a significant reduction in the time taken to learn the parameters and further demonstrated that they are close approximations of the maximum likelihood estimates.

8.6 Constructing a Hawkes Model for Social Media

The previous sections of this chapter introduced the theoretical bases for working with Hawkes processes. Section 8.2 and 8.3 gave the definitions and the basic properties of point processes, Poisson processes, and Hawkes processes. Section 8.4 and 8.5 respectively presented methods for simulating events in a Hawkes process and fitting the parameters of a Hawkes process to data. The aim of this section is to provide a guided tour for using Hawkes processes with social media data. We will start from customizing the memory kernel with a goal of predicting the popularity of an item. The core technique here is from a recent paper [[Mishra et al. 2016](#)] on predicting the size of a retweet cascade. In Section 8.6.1, we argue why

a Hawkes process is suitable for modeling the retweet cascades and we present the construction of the kernel function $\phi(t)$; in Section 8.6.2 we estimate real-life model parameters from Twitter data; in Section 8.6.3 we predict the expected size of a retweet cascade (i.e., its popularity).

8.6.1 A Marked Hawkes Process for Information Diffusion

We model *word of mouth* diffusion of online information: users share content, and other users consume and sometimes re-share it, broadcasting to more users. For this application, we consider each retweet as an event in the point process. We also formulate information diffusion in Twitter as a self-exciting point process, in which we model three key intuitions of the social network: *magnitude of influence* (tweets by users with many followers tend to get retweeted more), *memory over time* (that most retweeting happens when the content is *fresh* [Wu and Huberman 2007]), and *content quality*.

The event intensity function. A retweet is defined as the resharing of another person's tweet via the dedicated functionality on the Twitter interface. A retweet cascade is defined as the set of retweets of an initial tweet. Using the branching structure terminology introduced in Section 8.3, a retweet cascade is made of an *immigrant* event and all of its *offspring*. We recall the definition of the event intensity function in a Hawkes process, introduced in Equation (8.11):

$$\lambda(t) = \lambda_0(t) + \sum_{T_i < t} \phi_{m_i}(t - T_i). \quad (8.32)$$

$\lambda_0(t)$ is the arrival rate of immigrant events into the system. The original tweet is the only immigrant event in a cascade, therefore $\lambda_0(t) = 0, \forall t > 0$. Furthermore, this is modeled as a *marked* Hawkes process. The *mark* or magnitude of each event models the user influence for each tweet. The initial tweet has event time $T_0 = 0$ and mark m_0 . Each subsequent tweet has the mark m_i at event time T_i .

We construct a power-law kernel $\phi_m(\tau)$ with mark m :

$$\phi_m(\tau) = \kappa m^\beta (\tau + c)^{-(1+\theta)}. \quad (8.33)$$

κ describes the *virality*—or quality—of the tweet content, and it scales the subsequent retweet rate; β introduces a warping effect for user influences in social networks; and $1 + \theta$ ($\theta > 0$) is the power-law exponent, describing how fast an event is *forgotten*, while parameter $c > 0$ is a temporal shift term to keep $\phi_m(\tau)$ bounded when $\tau \simeq 0$. Overall, κm^β accounts for the magnitude of influence, and the power-

law kernel $(\tau + c)^{-(1+\theta)}$ models the memory over time. We assume user influence m is the observed number of followers obtained from Twitter API.

In a similar fashion, we can construct an exponential kernel for social media, based on the kernel defined in Equation (8.12):

$$\phi_m(\tau) = \kappa m^\beta \theta e^{-\theta\tau}. \quad (8.34)$$

We have experimented with this kernel and Figure 8.3(c) shows its corresponding intensity function over time for a real Twitter diffusion cascade. However, we have found that the exponential kernel for social media provides lower prediction performances compared to the power-law kernel defined in Equation 8.33. Consequently, in the rest of this chapter, we only present the power-law kernel.

8.6.2 Estimating the Hawkes Process

The marked Hawkes process has four parameters $\theta = \{\kappa, \beta, c, \theta\}$, which we set out to estimate using the maximum likelihood estimation technique described in Section 8.5. We can obtain its log-likelihood by introducing the marked memory kernel (8.33) into the general log-likelihood formula shown in Equation (8.29). The first two terms in Equation 8.35 are from the likelihood computed using the event rate $\lambda(t)$, the last term is a normalization factor from integrating the event rate over the observation window $[0, T]$:

$$\begin{aligned} \mathcal{L}(\kappa, \beta, c, \theta) = & \sum_{i=2}^n \log \kappa + \sum_{i=2}^n \log \left(\sum_{t_j < t_i} \frac{(m_j)^\beta}{(t_i - t_j + c)^{1+\theta}} \right) \\ & - \kappa \sum_{i=1}^n (m_i)^\beta \left[\frac{1}{\theta c^\theta} - \frac{(T + c - t_i)^{-\theta}}{\theta} \right]. \end{aligned} \quad (8.35)$$

Equation 8.35 is a non-linear objective that needs to be maximized. There are a few natural constraints for each model parameter, namely: $\theta > 0$, $\kappa > 0$, $c > 0$, and $0 < \beta < \alpha - 1$ for the branching factor to be meaningful (and positive). Furthermore, while the supercritical regimes $n^* > 1$ are mathematically valid, it will lead to a prediction of infinite cascade size—a clearly unrealistic outcome. We further incorporate $n^* < 1$ as a non-linear constraint for the maximum likelihood estimation. Ipopt [Wächter and Biegler 2006], the large-scale interior point solver, can be used to handle both non-linear objectives and non-linear constraints. For efficiency and precision, it needs to be supplied with pre-programmed gradient functions. Details of the gradient computation and optimization can be found in Mishra et al. [2016].

Section 8.5.2 warned about three possible problems that can arise when using maximum likelihood estimates with Hawkes processes: edge effects, squared computational complexity, and local minima. In this application, since we always observe a cluster of events generated by an immigrant, we do not have edge effects, i.e., missing events early in time. The computational complexity of calculating the log-likelihood and its gradients is $O(n^2)$, or quadratic with respect to the number of observed events. In practice, we use three techniques to make computation more efficient: vectorization in the R programming language, storing and reusing parts of the calculation, and data-parallel execution across a large number of cascades. With these techniques, we can estimate tens of thousands of moderately sized retweet cascades containing hundreds of events in a reasonable amount of time. Lastly, the problem of *local minima* can be addressed using multiple random initializations, as discussed in Section 8.5.2.

8.6.3 The Expected Number of Future Events

Having observed a retweet cascade until time T for a given Hawkes process, one can simulate a possible continuation of the cascade using the thinning technique presented in Section 8.4. Assuming a subcritical regime, i.e., $n^* < 1$, the cascade is expected to die out in all possible continuation scenarios. In addition to simulating a handful of possible endings, it turns out there is a close-form solution to the expected number of future events in the cascade over all possible continuations, i.e., the total popularity that the cascade will reach by the time it ends.

There are three key ideas for computing the expected number of future events. The first is to compute the expected size of a direct offspring to an event at T_i after time T ; the second is that the expected number of all descendent events can be obtained via the branching factor of the Hawkes process, as explained in Section 8.3.3. Lastly, the estimate of total popularity emerges when we put these two ideas together.

The number of future children events. In retweet cascades, the base intensity is null $\lambda_0(t) = 0$, therefore no new immigrants will occur at $t > T$. Equation (8.17) gives the expected size of a cluster of offspring associated with an immigrant. In the marked Hawkes process of Eq (8.32), each of the $i = 1, \dots, n$ events that happened at $T_i < T$ adds $\phi_{m_i}(t - T_i)$ to the overall event intensity. We can obtain the expectation of A_1 , the total number of events directly triggered by event $i = 1, \dots, n$, by integrating over the memory kernels of each event. The summation and integration are exchangeable here, since the effect of each event on future event intensity is additive:

$$\begin{aligned}
A_1 &= \int_T^\infty \lambda(t) dt = \int_T^\infty \sum_{t>T_i} \phi_{m_i}(t - T_i) dt \\
&= \sum_{t>T_i} \int_T^\infty \phi_{m_i}(t - T_i) dt = \kappa \sum_{i=1}^n \frac{m_i^\beta}{\theta (T + c - T_i)^\theta}. \tag{8.36}
\end{aligned}$$

The branching factor. The branching factor was defined in Equation (8.14) for an unmarked Hawkes process. We compute the branching factor of the marked Hawkes process constructed in Section 8.6.1 by taking expectations over both event times and event marks. We assume that the event marks m_i are *i.i.d.* samples from a power law distribution of social influence [Kwak et al. 2010]: $P(m) = (\alpha - 1)m^{-\alpha}$. α is an exponent that controls the heavy tail of the distribution, and it is estimated from a large sample of tweets. We obtain the closed-form expression of the branching factor (see Mishra et al. [2016] for details):

$$n^* = \kappa \frac{\alpha - 1}{\alpha - \beta - 1} \frac{1}{\theta c^\theta}, \quad \text{for } \beta < \alpha - 1 \text{ and } \theta > 0. \tag{8.37}$$

Total size of cascade. Putting both Eq (8.36) and Eq (8.37) together, we can see that each expected event in A_1 is anticipated to generate n^* direct children events, n^{*2} grand-children events, \dots , n^{*k} k -th generation children events, and so on. The calculation of geometric series shows that the number of all descendants is $\frac{A_1}{1-n^*}$. This quantity plus the observed number of events n is the total number of expected events in the cascade. See Mishra et al. [2016] for complete calculations.

$$N_\infty = n + \frac{\kappa}{(1 - n^*)} \left(\sum_{i=1}^n \frac{m_i^\beta}{\theta (T + c - t_i)^\theta} \right), \quad n^* < 1. \tag{8.38}$$

8.6.4 Interpreting the Generative Model

A Hawkes process is a generative model, meaning that it can be used to interpret statistical patterns in diffusion processes, in addition to being used in predictive tasks. Figure 8.3 presents a diffusion cascade about a *New York Times* news article with its corresponding intensity functions with the power-law and exponential memory kernels, respectively. Note that the top and lower two graphics are temporally aligned. In other words, each event occurs that causes a jump in the intensity function, i.e., increasing the likelihood of future events. Each jump is followed by a rapid decay, governed by the decay kernel $\phi_m(\tau)$, defined in Section 8.6.1. In terms of event marks, the cascade attracts the attention of some very well-followed accounts. The original poster (@screencrushnews) has 12,122 followers, and among

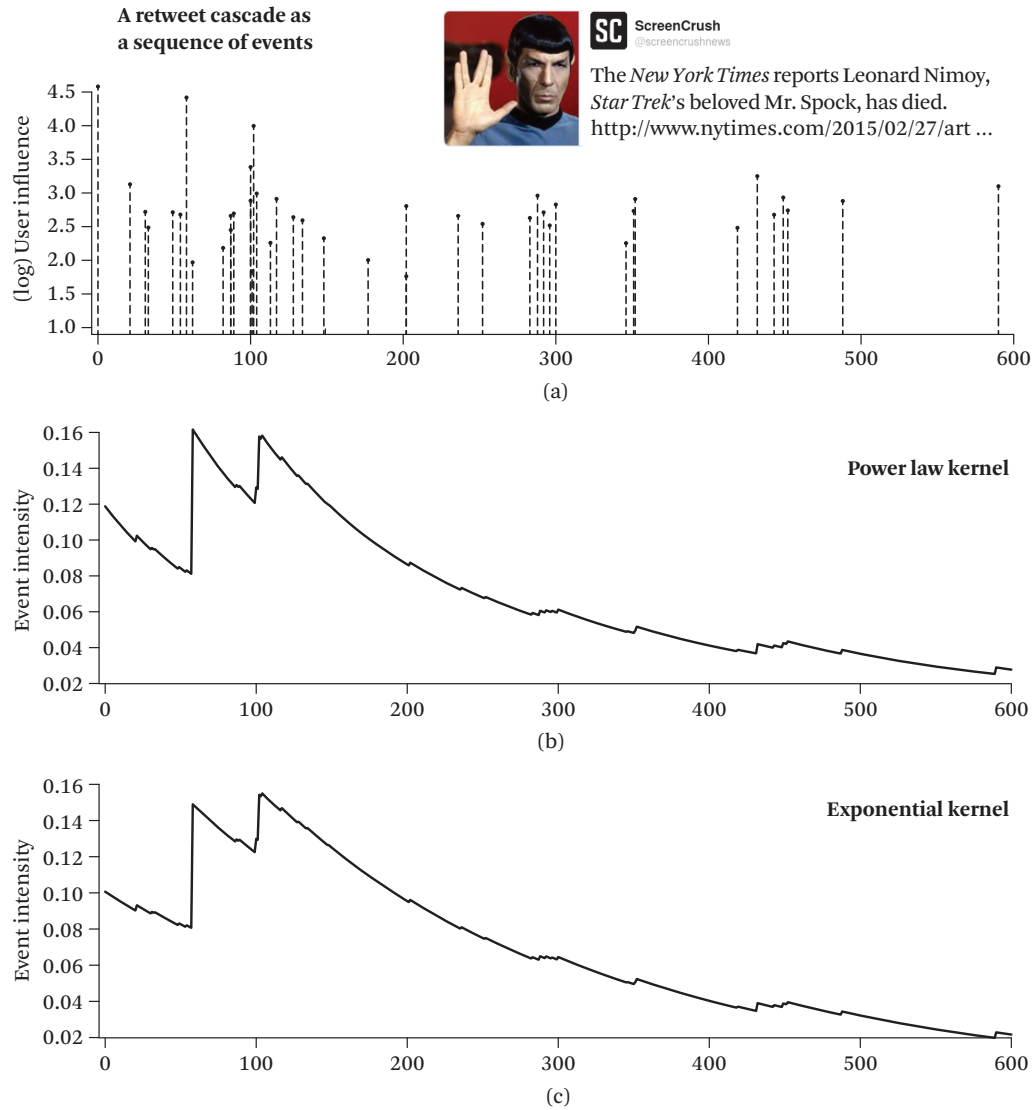


Figure 8.3 An example retweet cascade on a news article by the *New York Times* (<http://www.nytimes.com/2015/02/27/arts/television/leonard-nimoy-spock-of-star-trek-dies-at-83.html>). (a) Representation of the first 600 seconds of the retweet cascade as a marked point process; an event time corresponds to each (re)tweet. (b) Event intensity $\lambda(t)$ over time, assuming the point process to be a Hawkes process with power-law kernel. The maximum-likelihood model parameter estimates are $\{\kappa = 1.00, \beta = 1.01, c = 250.65, \theta = 1.33\}$ with a corresponding $n^* = 0.92$ and a predicted cascade size of 216. The true cascade size is 219. (c) The event intensity over time for the same event time series, when the point process is assumed to be a Hawkes process with the exponential kernel defined in Equation 8.34. The fitted parameters for this kernel are $\{\kappa = 0.0003, \beta = 1.0156, \theta = 0.0054\}$, the corresponding $n^* = 0.997$, and the predicted cascade size is 1603.

the users who retweeted, @TasteOfCountry (country music) has 193,081 followers, @Loudwire (rock) has 110,824 followers, @UltClassicRock (classic rock) has 99,074 followers, and @PopCrush (pop music) has 114,050 followers.

For popularity prediction, the cascade is observed for 10 minutes (600 seconds) and the parameters of the Hawkes process are fitted as shown in Section 8.6.2. The maximum-likelihood estimates of parameters with a power-law kernel are $\{\kappa = 1.00, \beta = 1.01, c = 250.65, \theta = 1.33\}$, with a corresponding $n^* = 0.92$. According to the power-law kernel, this news article has high content virality (denoted by κ) and large waiting time (c), which in turn leads to a slow diffusion: the resulting cascade reaches 1/4 its size after half an hour, and the final tweet is sent after 4 days. By contrast, most retweet cascades finish in a matter of minutes, tens of minutes at most. Using the formula in Equation (8.38), we predict the expected total cascade size $N_\infty = 216$; this is very close to the real cascade size of 219 tweets, after observing only the initial 10 minutes of the 4-day Twitter diffusion. When estimated with an exponential kernel, the parameters of the Hawkes point process are $\{\kappa = 0.0003, \beta = 1.0156, \theta = 0.0054\}$ and the corresponding branching factor is $n^* = 0.997$. This produces a very imprecise total cascade size prediction of 1603 tweets, largely due to the high n^* .

8.6.5 Hands-on Tutorial

In this section, we provide a short hands-on tutorial, together with code snippets required for modeling information diffusion through retweet cascades. A detailed version of the tutorial with example data and code is available at <http://github.com/s-mishra/featuredriven-hawkes>. All code examples presented in this section assume a Hawkes model with the power-law kernel. The complete online tutorial also presents examples that use an exponential kernel. All code was developed using the R programming language.

We start with visualizing in Figure 8.4 the shape of the power-law kernel (defined in Equation (8.33)) generated by an event with the mark $m = 1000$, and defined by the parameters $\kappa = 0.8$, $\beta = 0.6$, $c = 10$, and $\theta = 0.8$. The code for generating the figure is shown in Listing 8.1. Furthermore, we can simulate (Listing 8.2) the entire cluster of offspring generated by this initial immigrant event using the thinning procedure described in Section 8.4.1. The initial event is assumed to have occurred at time $t = 0$, and the simulation is run for 50 time intervals.

We now show how to estimate the parameters of a Hawkes process with a power-law kernel for a real Twitter diffusion cascade and how to estimate the total size of the cascade. The file `example_book.csv` in the online tutorial records the retweet diffusion cascade around a news article announcing the death of “Mr. Spock” shown in Figure 8.3. Figure 8.3(a) depicts the cascade as a point process:

```

## initial event that starts the cascade (i.e. the immigrant)
## mark = 1000 at time t = 0
event <- c(mark = 1000, time = 0)
## the timepoints for which we compute the kernel function
t <- seq(from = 0, to = 100, length=1000)
## set the parameters of the kernel
K <- 0.8
beta <- 0.6
c <- 10
theta <- 0.8

## compute the Power Law Kernel
## call the kernelFunction to get the values
values.PL <- kernelFct(event = event, t = t, K = K, beta = beta, c = c,
                      theta = theta, kernel.type='PL')

## plot the obtained kernel
plot(x = t, y = values.PL, type = "l", col = "blue",
     xlab = "", ylab = "", main = "Power-law memory kernel over time")

```

Listing 8.1 Code for computing the power-law kernel function, generated by an event with mark 1000.

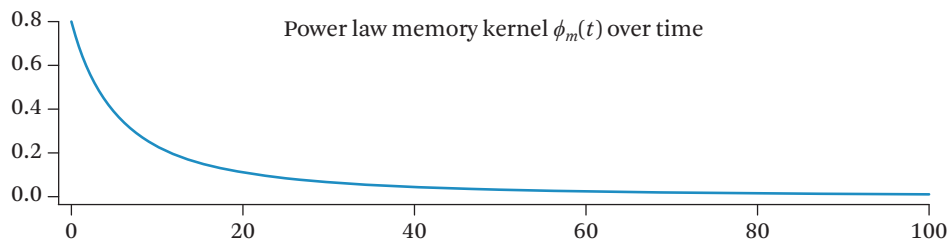


Figure 8.4 Graphic obtained by running the code in Listing 8.1: The power-law kernel over time, generated by an event with the mark 1000.

the tweet posting times are the event times, whereas the numbers of followers of the users emitting the tweets are considered the event marks. The code in Listing 8.3 reads the CSV file and performs a maximum likelihood estimation of the Hawkes process parameters, based on the events in the cascade having occurred in the first 600 seconds (10 minutes). With the obtained estimates for model parameters, we can predict (using the code in Listing 8.4) the total size of the diffusion cascade.


```
## simulating an event series
events <- generate_Hawkes_event_series(K = K, beta = beta, c = c,
                                     theta = theta, M = event["mark"],
                                     Tmax = 50)
```

Listing 8.2 Simulation using the thinning method (see Section 8.4.1) of an entire cluster of offspring, generated by the immigrant defined in Listing 8.1 ($m = 1000$, $t = 0$).

```
## read the real cascade provided in the file "example.csv"
real_cascade <- read.csv(file = 'example_book.csv', header = T)
## retain only the events that occurred in the first 600 seconds (10min).
## These will be used for fitting the model parameters.
predTime <- 600
history <- real_cascade[real_cascade$time <= predTime, ]
## removing the first column, which is event index
## retaining column 2 (event mark) and column 3 (event time).
history <- history[ , 2:3]
## call the fitting function, which uses IPOPT internally. The fitting
## algorithm requires an initial guess of the parameters. This can be
## a random point within the domain of definition of parameters.
startParams <- c(K = 1, beta = 1, c = 250, theta = 1)
result <- fitParameters(startParams, history)
```

Listing 8.3 Load the information about the real tweet cascade from Figure 8.3, and fit parameters using the events observed in the first 600 seconds.

```
## Using the fitted model parameters, we call getTotalEvents to get
## predictions.
prediction <- getTotalEvents(history = history, bigT = predTime,
                             K = result$solution[1],
                             beta = result$solution[2],
                             c = result$solution[3],
                             theta = result$solution[4])
## The "prediction" object contains other values, such as
## the branching factor (nstor) and A1
nPredicted = prediction['total']
```

Listing 8.4 Predict the total size for the Twitter cascade shown in Figure 8.3, using the parameters fitted as in Listing 8.3.

8.7 Conclusion

This chapter provided a gentle introduction for the Hawkes self-exciting process. We covered the key definitions of point processes and Hawkes processes. We introduced the notion of event rate, branching factor, and the use of these quantities

to predict future events. We described procedures for simulating a Hawkes process, and derived the likelihood function used for parameter estimation. We also included a practical example for estimating a Hawkes process from retweet cascades, along with code snippets and online notebook. Where applicable, we have included discussions of the point-process literature. The goal of the materials above is to provide the fundamentals to researchers who are interested in formulating and solving application problems with point processes. Interested readers are invited to explore more advanced materials, including: alternative inference algorithms such as using expectation-maximization, sampling, or moment matching; flexible specifications and extensions of self-exciting processes such as multivariate mutually exciting Hawkes processes, and doubly stochastic processes, to name a few.



Utilizing Implicit User Cues for Multimedia Analytics

Subramanian Ramanathan (International Institute of Information Technology, Hyderabad),
Syed Omer Gilani (National University of Sciences and Technology, Islamabad),
Nicu Sebe (Università di Trento)

9.1 Introduction

As humans, we routinely interact with media in the form of text, speech and audio, image, video, and new media such as Internet webpages and virtual/augmented reality, as well as other humans. During the course of these interactions, we convey our understanding of (or alternatively, response to) the environment via a variety of behavioral signals. These signals can be exhibited either consciously or sub-consciously by the perceiver (user), and may be explicitly or implicitly conveyed to the external world. Speech is an example of a behavioral signal that is consciously expressed by a person, and whose conveyance is conspicuous to the outside world. Gazing behavior is another example. While we may change our head/body orientation sub-consciously to focus on a counterpart during face-to-face conversations, such changes are apparent to the outside world and head/body pose cues have been extensively used in security and behavioral analytics (see [Murphy-Chutorian and Trivedi \[2009\]](#) and [Subramanian et al. \[2013\]](#) for related work). This chapter expressly focuses on the extraction and utilization of implicit behavioral cues such as eye movements and brain signals, which are exhibited sub-consciously by the user and are invisible to the external world, for multimedia analytics.

Multimedia understanding and computer vision systems have long relied on explicit and implicit user cues for input and feedback. Retrieval has greatly benefited from user meta-data in the form of textual tags. Likewise, the ESP game [von Ahn and Dabbish 2004] and the LabelMe database [Russell et al. 2008] have greatly contributed to visual object recognition. Although advances in deep learning [Krizhevsky et al. 2012] have significantly facilitated automated multimedia understanding, deep networks still cannot reliably substitute for human intelligence.¹ This is partly because human understanding of scene semantics is superior to computational models, and the difference in information interpreted by a human from data, and what is computationally extractable from the same is called semantic gap [Smeulders et al. 2000].

Annotations in the form of spatial markings, keywords, or sentence captions have helped media analytic approaches bridge the semantic gap. Nevertheless, manually annotating large databases is tedious and expensive, which has spurred alternative approaches employing implicit and rapid-eye or brain-based annotations [Aloimonos et al. 2012, Papadopoulos et al. 2014, Shenoy and Tan 2008, Wang and Dey 2009] in the past decade. Examples of how implicit user annotations have been successfully utilized for multimedia analytics include (i) scene and (ii) emotion understanding from images and videos.

Scene understanding involves a number of sub-tasks such as (i) segmentation and recognition of individual scene objects, (ii) discovering object interactions therefrom, (iii) 3D spatial understanding of the scene, and (iv) making high-level inferences (e.g., textual description and question answering, estimating the level of emotion) from the scene content. Extensive labeled training data is required to accomplish each sub-task, and implicit annotations acquired via eye-trackers and wearable Electroencephalogram (EEG) devices have been employed to augment visual features. Two recent studies that use eye fixations for object detection from the PASCAL Visual Object Classes [Everingham et al. 2014] dataset are Yun et al. [2013] and Papadopoulos et al. [2014]. Brain-based annotations in the form of EEG signatures have also been exploited for object recognition. The discriminability of EEG signatures for disparate object classes is shown by Shenoy and Tan [2008], while Kapoor et al. [2008] augment visual features with EEG features to achieve superior object recognition than with either type alone. Other work in this regard is that of Wang and Dey [2009], which combines the strengths of human visual perception (high accuracy but limited throughput) and computer vision (high throughput but

1. <http://www.nytimes.com/2016/09/20/science/computer-vision-tesla-driverless-cars.html>

limited accuracy). Here, EEG-based object annotations are combined with visual features to characterize target objects, and this knowledge is propagated over a large dataset using a visual similarity graph for large-scale object recognition.

Emotion recognition has been another problem of interest to the multimedia community for some time. Ever since the need for indexing videos based on the emotions they evoke was identified by [Hanjalic and Xu \[2005\]](#), a number of researchers have attempted to recognize emotions by examining audio [[Lee and Narayanan 2005](#)], visual [[Lucey et al. 2010](#)], and audio-visual content [[Sebe et al. 2006](#)]*—*these are termed content-based emotion recognition (ER) approaches. However, these approaches only met with limited success as there exists a gap between emotion conveyed by the content, and the emotion induced in the viewer*—*e.g., a crying comedian can invoke laughter in the audience. This gap consequently prompted the development of user-centered ER approaches, which monitor viewers' behavioral cues to predict the emotion evoked by the stimulus. Here again, initial research focused on identifying cues such as facial expressions [[Joho et al. 2011](#)], which denote a conscious manifestation of emotions and are therefore suppressible. User-centered ER later evolved to measure brain and physiological activity such as EEG and magnetoencephalogram or MEG signals, heart and respiration rate, skin conductance, and muscle movements, which are sub-conscious emotional manifestations and thereby hard to suppress. Examples of recent ER research examining implicit physiological user responses include DEAP [[Koelstra et al. 2012](#)], MAHNOB [[Soleymani et al. 2012](#)], and DECAF [[Abadi et al. 2015](#)].

In relation to the aforementioned problems, this chapter discusses

- How eye movements, in the form of fixations and saccades, enable inference of scene semantics as demonstrated by [[Subramanian et al. 2011](#)] (Section 9.2). Saccades (ballistic eye movements) and fixations (stationary phases where visual information is absorbed) can reveal information such as an interaction between two scene objects (e.g., two men talking), or the emotional intensity of faces (mildly intense vs. highly intense facial expression). We also present recent work on gaze-assisted object detection by [Gilani et al. \[2015\]](#) (Section 9.3), where eye fixations serve as implicit annotations denoting the spatial locations of target objects in the scene; Gilani and colleagues describe how visual descriptors extracted around fixated locations improve object recognition performance.
- Implicit cues can convey information regarding the media with which users interact, and regarding the users themselves. In this regard, we present recent work by [Wache et al. \[2015\]](#) (Section 9.4) on recognizing users' emotions

and personality type based on physiological responses observed during affective clip viewing. Experiments to recognize the “big-five” personality traits [Perugini and Di Blas 2002] suggest that personality differences are better revealed on examining user responses to clips conveying similar emotions.

The chapter will conclude by reflecting on how monitoring implicit user cues is important for related areas such as safety and security, interaction design, and big-data visualization (Section 9.5).

9.2 Inferring Scene Semantics from Eye Movements

For over a decade, many researchers have studied human visual attention in order to develop computational saliency models that can predict regions or objects of interest in images and videos. Following the bottom-up saliency proposition that attributed visual attention to low-level visual features such as intensity, color, and orientation (see Itti and Koch [2000] and Valenti et al. [2009] for examples), later work such as that by Judd et al. [2009] and Subramanian et al. [2010] has noted that top-down or meaningful factors such as faces, cars, and emotional scene objects also play an important role in attracting viewers’ attention. Nevertheless, most of these approaches purely focus on scene fixations, ignoring saccadic information that captures the temporal fashion in which the scene is scanned. In one of the first papers to examine saccadic patterns and model them in a computational setting for scene understanding, Subramanian et al. [2011] show how saccades are useful for (i) discovering interactions between scene objects (typically described using transitive verbs as in *Man reads a book*), and (ii) for distinguishing between high- and low-intensity facial expressions.

9.2.1 Discovering Object Interactions via Saccades

In order to compile ground truth concerning whether an image depicts an interaction or not, Subramanian et al. [2011] gathered succinct text captions for 110 social scenes, where humans are seen performing day-to-day actions. These 110 images were compiled from the MIT1003 [Judd et al. 2009] and NUSEF [Subramanian et al. 2010] eye-tracking datasets, and eye-movement recordings compiled from at least 13 viewers are available for each image. Caption descriptions for each of the 110 images were obtained from 12–20 (independent) observers, and the named frequency of scene objects and object interactions were computed from the descriptions. For 45 images, over 80% of the descriptions contained an interaction verb—these scenes were deemed to be interactive. For 37 images, fewer than 30%

viewers reported any form of interaction—these scenes were considered to be non-interactive.

The eye-gaze patterns for the interactive and non-interactive scenes were then examined, and Figure 9.1 presents an exemplar of each type, and corresponding gaze patterns. The (collective) eye fixations are plotted in yellow, and the fixations are clustered using the mean-shift-based, multi-scale fixation clustering approach [Santella and DeCarlo 2004] to form clusters denoted by color-coded and numbered ellipses. Arrows denote saccades from one fixation cluster to another, and are color-coded based on the originating cluster—e.g., saccades originating from the red cluster numbered “1,” and ending in the green cluster numbered “2” for the interactive scene are denoted by the red arrow. Furthermore, thickness of each arrow represents the likelihood of a saccade from the source to the sink fixation cluster.

If we compare saccades for interactive and non-interactive scenes, a crucial difference is that object interactions such as read, talk, point, etc. are consistently characterized by vacillating saccades between interacting objects. Psychology literature provides support to this phenomenon: such interactions are characterized by the mutual orientation of interacting objects, and humans are instantly able to determine and follow the direction of others’ attention [Kuhn et al. 2009]. Also, short-term or episodic scene memory causes viewers to semantically re-fixate on interesting objects. These two phenomena jointly lead to the occurrence of vacillating saccades between interacting objects. However, such vacillating saccades are unlikely to occur for non-interactive scenes as shown in the Figure 9.1 (right) example, where only uni-directional saccades are noticeable. The authors of this example automatically detect object interactions via saccadic behavior, by



Figure 9.1 (from left to right) Exemplar image showing interaction between objects (two men talking to one another). Corresponding fixations (yellow dots) and gaze clusters: arrows denote saccades from one fixation cluster to another, and are color coded based on the originating cluster. Exemplar non-interactive image (two girls posing for the camera while eating). Corresponding fixations and saccades. Figure best viewed under zoom. (Adapted from Subramanian et al. [2011])

computing the ratio of saccading probabilities for object pairs l, m . [Subramanian et al. 2011]. More specifically, the saccading probability from object l to object m is defined as $P(m/l) = \frac{|S_{l,m}|}{|F_l|}$, where $|S_{l,m}|$ and $|F_l|$ respectively denote number of saccades from l to m , and the number of fixations on object l . Then, the ratio of back-and-forth saccades is computed as $\eta_s = \min(\frac{P(m/l)_s}{P(l/m)_s}, \frac{P(l/m)_s}{P(m/l)_s})$ at multiple spatial scales for clustering fixations indexed by s . The spatial scale s corresponding to $\text{argmin}(\eta_s)$ (implying most symmetric saccading behavior) is then deemed as the scale at which an interaction is detected.

The above approach to detect object interactions relies purely on eye movement patterns, and explicit correspondence between fixation clusters and scene objects is not computed based on the implicit assumption that fixations tend to fall on salient scene objects. The authors attempt to identify objects on which eye fixations fall via object detectors. An improvement over this methodology, where explicit correspondence between fixation clusters and scene objects is computed via joint clustering of fixations and visual entities via a graph structure, is presented by Sugano et al. [2013]. These authors define a graph structure over both eye fixations and image super-pixels (partitions obtained on oversegmenting the image), and this graph exploits spatial correlation between fixation clusters and visual entities. Joint clustering is accomplished through an iterative labeling of the graph nodes through energy minimization, and this methodology can automatically determine the optimal number of clusters that are associated with both fixations and visual entities. Interested readers may refer to Sugano et al. [2013] for more details.

9.2.2 Differentiating Low-intensity and High-intensity Facial Expressions via Eye Movements

Subramanian et al. [2011] have also shown how eye movements can be combined with visual information to distinguish between low-intensity and high-intensity facial expressions. Again, in order to define mildly intense and highly intense facial emotions, 12 viewers were asked to rate the intensity of the portrayed facial expression in 60 images for which eye movement recordings were available. All the images contained an upright and roughly frontal face captured at high resolution as shown in Figure 9.2. Observers were required to say whether they perceived the facial emotion as positive, neutral, or negative (termed valence in psychological literature). If they considered the emotional valence to be positive or negative, they were required to rate the emotional intensity on a Likert scale of 1–7. However, if an observer perceived a face to be neutral, the emotional intensity score was assigned to 0. The median of the observer scores (M_S) was used as a threshold to determine whether a face was perceived as mildly or highly expressive, and faces

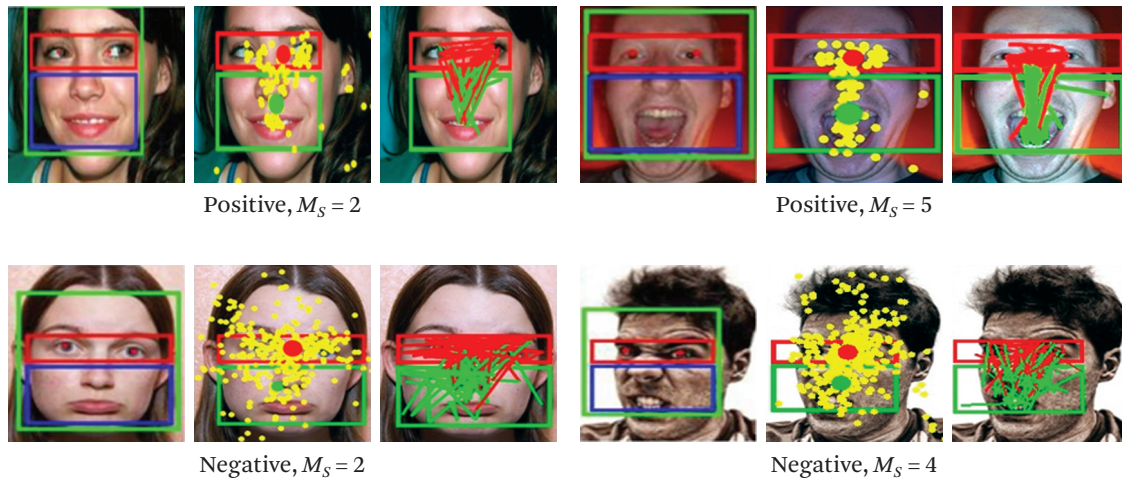


Figure 9.2 Two positive emotional faces, and their fixation and saccade plots (top row: from left to right). Two negative emotional faces and their gaze patterns (bottom row: from left to right). Median observer score M_S denoting perceived emotional intensity by the observer is shown below for each face. (Adapted from [Subramanian et al. \[2011\]](#))

with $M_S > 3$ (28/60) were considered highly emotional while the rest were deemed to be mildly emotional. Four emotional faces (two positive and two negative), and their corresponding M_S scores, are presented in Figure 9.2.

Eye movements on these face images were then studied upon automatically determining region-of-interest rectangles via the Viola-Jones face detector [[Viola and Jones 2004](#)] and the neural-network based Rowley eye detector [[Rowley et al. 1998](#)] (more robust and accurate fiducial face feature detection methods such as [[Xiong and De La Torre 2013](#), [Baltrušaitis et al. 2016](#)] exist today). The green, red, and blue rectangles in Figure 9.2 respectively denote the face, eye (upper face half), and nose+mouth (lower face half) regions—detected eye centers are shown using red dots. The second and third images corresponding to each face depict eye movements: fixations are shown using yellow dots, while saccades originating from the upper and lower face halves are denoted using red and green lines, respectively. Red and green circles in the fixation plots denote centroids of the upper/lower face half fixations, while the size of these dots indicate, density of fixations in the respective half.

Eye tracking data for emotional faces reveals that eyes are the most salient face regions, and a majority of the fixations appear around the eyes for mildly/moderately expressive faces. However, with increase in the emotional intensity, more

and more fixations appear in the lower face half when considerable deformations are observed around the nose and mouth regions. Overall, fixations are more frequent and longer on the interesting face regions, and saccades are likelier to occur from a less salient to a more salient face region. A linear SVM-based binary classification experiment to detect low-intensity and high-intensity facial emotions based on eye movement data achieved 67.5% and 72% recognition accuracy with fixation and saccade-based features, indicating the importance of saccades toward revealing interesting scene entities. The next section discusses how fixations serve as implicit annotations for object locations, and contribute to the improved recognition of scene objects.

9.3 Eye Fixations as Implicit Annotations for Object Recognition

In Section 9.1, we remarked how computers, despite research developments, are unable to understand the visual world like humans do. Scene understanding involves recognition (detecting presence or absence of a certain object class), localization (detection as well as demarcation of the object location via a bounding box) and segmentation (precise estimation of an object's spatial extent termed foreground, with respect to remainder of the scene known as background) of scene objects—and each of these problems remain unsolved at this time. The Pascal Visual Objects Classes (VOC) challenge [Everingham et al. 2014] was initiated in 2005 with the objective of offering a large and publicly available dataset to further research in these areas. Of late, a number of researchers including Yun et al. [2013] and Papadopoulos et al. [2014] have employed eye fixations as implicit annotations to facilitate model synthesis from the VOC dataset. Nevertheless, some disagreement exists between researchers as to whether fixations obtained during free-viewing (examining visual scenes without any pre-defined task) or visual search (where the viewer is explicitly instructed to look out for instances of certain target classes) provide optimal annotations.

To this end, Gilani et al. [2015] examine how free-viewing (FV) and visual search (VS) tasks affect fixations on target scene objects. They compile the PET or Pascal animal classes Eye Tracking database,² which comprises eye movement recordings for the bird, cat, cow, dog, horse, and sheep training+validation (or trainval) sets from the VOC2012 dataset. In PET, eye movements are recorded for both FV and VS conditions from 40 observers who viewed each image for 2 seconds, so that four user gaze patterns are available per image and condition. The six animal object classes,

2. <http://vintage.winklerbros.net/pet.html>

out of a total of 20 classes in the VOC2012 image set, were chosen because: (i) animal classes such as cats, dogs, and birds are challenging to detect using supervised learning methods owing to large intrinsic shape and textural variations [Parkhi et al. 2011]; and (ii) it would be advantageous to exploit human visual perception for recognizing these object classes, as psychophysical studies (e.g., Judd et al. [2009]) have noted our propensity to sense animals that are both predators and prey. We now describe this study in detail.

9.3.1 Materials and Methods

Stimuli and Participants. 4135 images from the Pascal VOC2012 dataset [Everingham et al. 2012] were used for this study. These images contained one or more examples of the bird, cat, cow, dog, horse, and sheep animal object classes, and also humans. 2549 images contained exactly a solitary instance of the target classes, while 1586 images contained either multiple examples of the animal classes, or a mixture of animals and humans. Specifically considering images with multiple animal instances, the mean number of animals per image was 3.1 ± 2.7 , which covered 0.45 ± 0.28 of the image area based on rectangular bounding box annotations available with the VOC dataset. 40 university students (18–24 years, 22 males) participated in the experiments.

Experimental Procedure. The eye-tracking experiment lasted for about 40 minutes, and was performed over two blocks with a short break in-between. Participants viewed about 800 images in these two blocks, with each image displayed for 2 s, and followed by a blank screen for 0.5 s. All observers were instructed to free-view images in the first block, and to “detect all animals in the scene” (visual search) for the second block. The VS block was always scheduled after the FV block to avoid a viewing bias. To maximize viewer engagement, a few distractor images that did not contain even a single instance of the target animal classes were also included in each block. The set of images in the FV and VS blocks was counterbalanced across viewers, and the images within each block were shown randomly. All images were shown at 1280×1024 pixel resolution on a 17" LCD monitor placed about 60 cm from the viewer. Eye movements were recorded using a Tobii desktop eye tracker, with a 120-Hz sampling frequency and accuracy of within 0.4° visual angle upon calibration.

9.3.2 Eye Movement Behavior During Free-viewing and Visual Search

Fixation density maps (or heat maps) qualitatively reveal scene regions that capture the viewer’s visual attention, and also provide a measure of attention dispersion

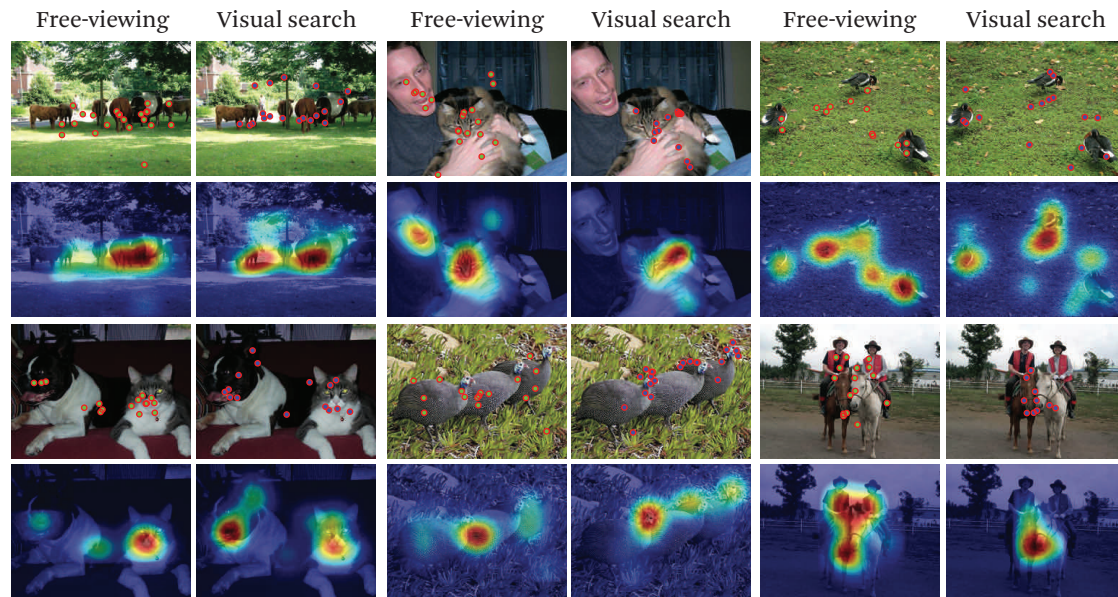


Figure 9.3 Recorded eye fixations and fixation density maps for the six animal categories (two per column) considered in PET.

in the scene. Figure 9.3 presents fixation density maps for the six PET animal classes. Considering pairs of rows, the top row shows eye fixations made by viewers for the FV and VS tasks, while the bottom row presents the corresponding heat maps obtained on convolving fixated scene locations with a Gaussian kernel of 2° visual angle bandwidth. Visual inspection of Figure 9.3 reveals that relatively similar density maps are obtained for both the FV and VS conditions, and that faces attract maximal visual attention.

Bounding box annotations for the animal classes are available as part of the Pascal VOC dataset. Utilizing bounding box coordinates in scenes containing multiple instances of the target classes, and specifically considering the first five fixations made by each viewer,³ the authors computed:

1. the proportion of fixations falling within the bounding rectangles, which was found to be 0.33 ± 0.26 for both the FV and VS conditions

3. The first five fixations were considered since they best convey observer intent.

2. the proportion of animal instances that had at least one fixation falling on them, which was again equal to 0.73 ± 0.26 for both conditions
3. the time taken by each user to fixate on at least half of the target objects, known as saccadic latency (Saccadic latency for VS (0.40 ± 0.34) was found to be lower than for FV (0.48 ± 0.35), and an independent *t*-test revealed that this difference was highly significant ($p < 0.000001$.)
4. the average duration for which each animal object was fixated in the two conditions—the mean durations for the FV and VS conditions were 0.51 ± 0.26 and 0.47 ± 0.28 , and this difference was again found to be highly significant ($p < 0.000001$)

Finally, the proportion of fixations falling within the target rectangles was found to be roughly equal in the FV and VS conditions for each of the animal classes, as seen in Figure 9.4(a).

Moving on from overall statistics, fine-grained analysis of gaze patterns was performed in the FV and VS conditions. Examining the duration of each fixation made by users in the two conditions, the first few fixations were longer and followed by progressively shorter fixations, as seen in Figure 9.4(b). Consistent with the overall statistics, per-fixation durations for FV were consistently higher as compared to VS up to the sixth fixation, while subsequent fixations were similar in duration.

Recurrence Quantification Analysis (RQA). Anderson et al. [2013] examines the dynamics of a single scan path, and is useful for measuring the compactness in viewing behavior. Compactness is quantified in terms of recurrence—the number (or proportion) of fixations repeated on previously fixated locations; determinism—proportion of recurrent fixation sequences representing repeated gaze trajectories; laminarity—denoting the proportion of fixations in a region being repeatedly fixated, and center of recurrent mass (CROM)—which measures the time interval between recurrent fixations. Based on RQA, viewing behavior during VS was found to be significantly more compact than during FV, in terms of all the four considered measures ($p < 0.01$ with two-tailed *t*-tests). In multimatch analysis [Foulsham et al. 2012], the sequence of fixations made by a viewer is treated as a vector denoting the scan path. Then, the set of scan paths obtained for two different conditions are processed to quantify the inter-conditional differences via saccade shape, length and direction, fixation locations and durations. The algorithm returns a similarity score in the range [0–1]. Figure 9.4(c) illustrates similarity among the above measures for the FV and VS tasks (considering all observers and images). Gaze behavior in both

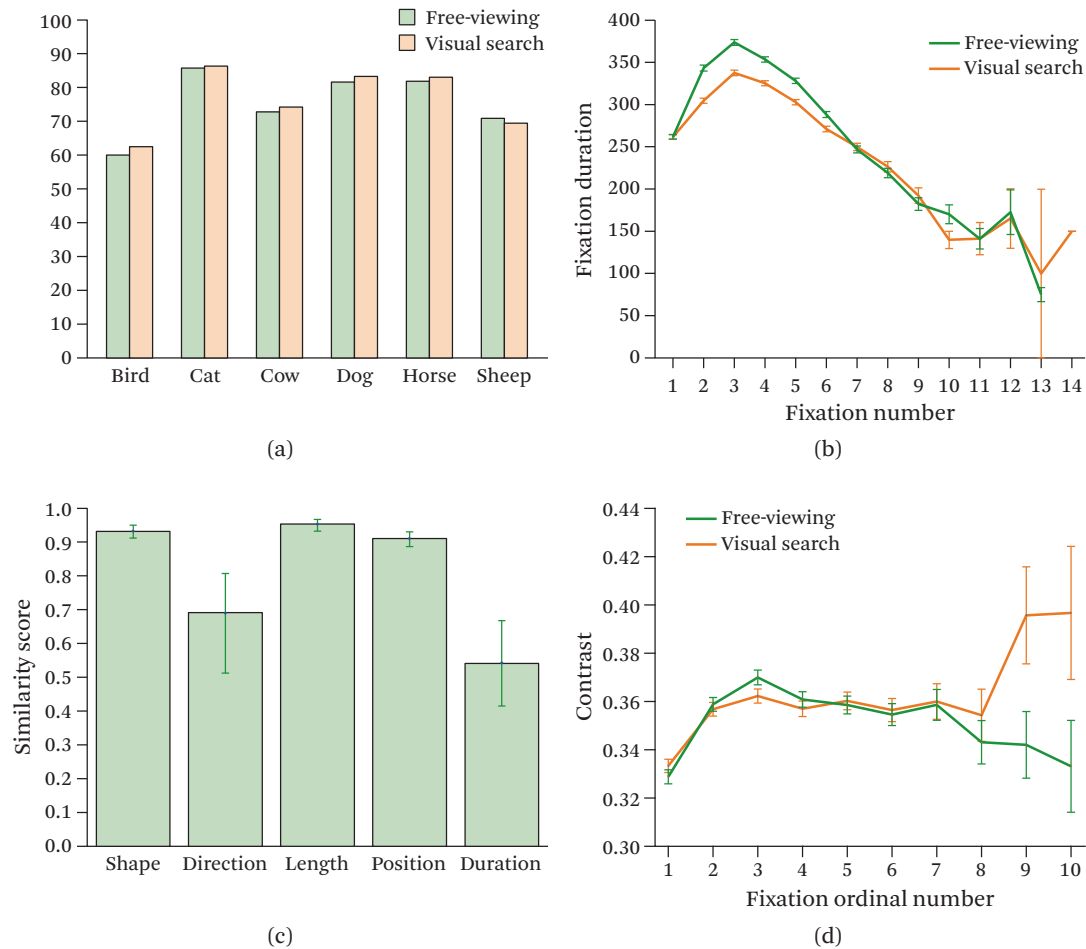


Figure 9.4 (a) Class-wise distribution of fixation proportions (in %) on animal object classes. (b) Comparing per-fixation durations ($\mu \pm \sigma$ shown) for the FV and VS conditions. (c) Multimatch similarity scores between the FV and VS conditions with standard errors plotted. (d) Contrast values at fixated locations as a function of fixation order (ordinal fixation number). Figure best viewed under zoom. (Adapted from [Gilani et al. \[2015\]](#))

conditions is similar in a number of respects, and differences are observed only with respect to saccade direction and fixation duration.

The fact that luminance contrast guides visual attention for natural scenes is well known in the literature [[Krieger et al. 2000](#), [Parkhurst and Niebur 2003](#), [Peters](#)

et al. 2005], and many computational models [Itti et al. 1998, Rajashekar et al. 2008] use it as a key feature for computing saliency maps. Hypothesizing that FV is more likely to be guided by bottom-up factors as compared to VS, contrast analysis was performed to quantify any observable differences in the fixated location properties, within and between the FV and VS conditions. The contrast C is defined as the local standard deviation in pixel intensities of square patches centered at the N fixation locations, normalized by the image mean intensity as defined below. We considered local patches of size 20×20 pixels,

$$C = \bar{I}^{-1} N^{-1} \sum_k \sqrt{\sum_{(i,j) \in \prod_k} (I_{i,j} - \bar{I}_k)^2},$$

where $I_{i,j}$ is the pixel intensity at location (i, j) in the image I , \bar{I}_k is the mean intensity of the patch \prod_k centered at k , and \bar{I} is the global mean intensity.

Within the FV and VS conditions, the contrast measure was computed across observers and images according to ordinal fixation number (1, 2, 3, . . .). A repeated-measures ANOVA for the FV and VS conditions was performed to determine if there were (contrast-based) differences among the locations fixated over time, as given by the ordinal fixation numbers. ANOVA revealed the main effect of contrast on viewing behavior over time for FV ($F(6, 27818) = 19.34, p < 0.001$). However, the impact of contrast on fixation behavior was not significant for the VS condition ($F(6, 23367) = 0.46, \text{n.s.}$). This implies that while viewers start by fixating at low-contrast locations in both conditions, they subsequently move to locations with higher-contrast during FV, consistent with a bottom-up visual attention model. While higher contrast regions are also chosen in the VS condition, the relative difference in contrast is not significant as visual attention is guided by the semantics related to the search prior. A post-hoc multiple comparison test, which enables fine-grained examination of contrast influence on fixation behavior, showed a consistent difference in contrast between ordinal fixation numbers for the FV condition. More specifically, the mean contrast value significantly varies with respect to the first fixation for five subsequent fixations in the FV condition. However, none of the subsequent fixations correspond to a significantly different contrast value for VS.

To examine overall differences in fixating behavior between the two conditions, contrast statistics between the two conditions were compared. To this end, fixations across all viewers and images were grouped according to ordinal number as earlier, and the mean contrast values for the FV and VS conditions were compared as shown in Figure 9.4(d). Clearly, viewers select locations of higher contrast in the FV condition ($\mu = 0.36 \pm 0.01$) as compared to VS ($\mu = 0.35 \pm 0.01$), which confirms

the hypothesis that low-level sensory information guides visual attention during free-viewing, while semantics drive attention during visual search.

9.3.3 Discussion

Examination of viewing behavior in the FV and VS conditions reveals that observers focused on target animal objects more quickly (lower saccadic latency) and showed greater urgency in scanning the scene (lower overall fixation and per-fixation duration) during visual search. The recurrence quantification and multimatch analyses revealed differences in terms of saccade direction, and the compactness of fixated locations in the two tasks. The fact that scan paths were found to be more compact during VS suggests that viewers tended to recurrently traverse (what they perceived as) informative scene regions instead of being guided by low-level scene elements. This hypothesis was further confirmed via contrast analysis. Examining contrast values at fixated locations within the FV and VS conditions, contrast differences over time were significant during FV. Also, analyzing contrast values over the two conditions, locations fixated during FV had consistently higher contrast as compared to locations fixated over time during VS. Collating the above results, we make the following comments regarding the suitability of FV and VS for priming fixation behavior.

- The visual search task motivates viewers to preferentially focus on designated targets (animals), and traverse the scene with more urgency in comparison to free-viewing.
- There is little difference between the FV and VS conditions in terms of the attention devoted to the target objects, as the proportion of fixations observed on target objects as well as the proportion of target objects fixated are very similar in both. Therefore, one would normally not expect much difference in performance if the fixated locations were to be used for model training. Nevertheless, the target specificity and compactness observed for the search task indirectly facilitates the learning of purer (less noisy) signatures for object recognition as described next.

9.3.4 Object Recognition with Fixation-based Annotations

The bag-of-words model⁴ is extremely popular for object recognition. However, it does not encode any spatial information. Spatial pyramid histogram representation is a more sophisticated approach in this respect, as it includes spatial information

4. http://en.wikipedia.org/wiki/Bag-of-words_model

for object recognition and consists of two steps: coding and pooling. The coding step involves point-wise transformation of descriptors to a representation better adapted to the task, while the pooling step combines outputs of several nearby feature detectors to synthesize a local or global bag of features. Pooling is employed to (i) achieve invariance to image transformations, (ii) arrive at more compact representations, and (iii) achieve better robustness to noise and clutter. However, spatially pooled regions are naively defined and spatial pyramid match [Yang et al. 2009] works by partitioning the image into increasingly finer sub-regions, and computing histograms of local features within each sub-region. These regions are usually squares which are sub-optimal for object recognition due to the inclusion of unnecessary background information. Given that viewers tend to fixate on salient (foreground) regions, fixated locations provide a valuable cue regarding the image features to be used for learning. Therefore, Gilani et al. [2015] pool features from regions around fixated locations instead of the entire image. Figure 9.5 illustrates the architecture of fixation-based feature pooling.

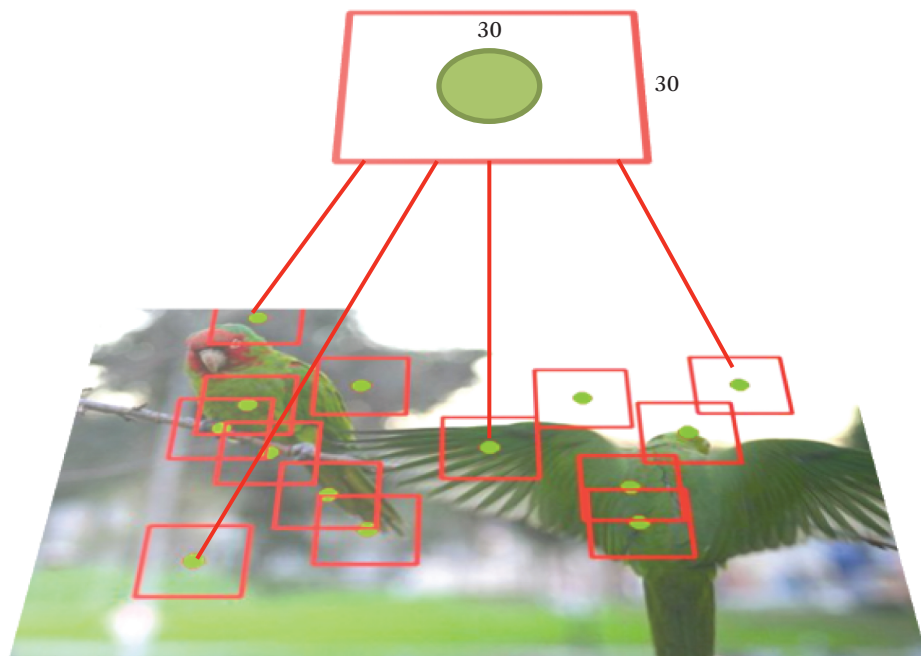


Figure 9.5 Feature pooling based on fixated locations: Green dots denote eye fixations. SIFT features are pooled within a window of size 30×30 around the fixated location. (Adapted from Gilani et al. [2015])

Linear spatial pyramid matching with sparse coding [Yang et al. 2009] has been successfully employed for object recognition. Sparse coding has been shown to find succinct representations of stimuli, and model data vectors as a linear combination of a few dictionary codewords. Adopting sparse coding in the coding stage, different pooling strategies are evaluated in Gilani et al. [2015]. Sparse coding is defined as:

$$\begin{aligned} \min_{\mathbf{D}, \mathbf{C}} \|\mathbf{X} - \mathbf{CD}\|_F^2 + \lambda_1 \|\mathbf{C}\|_1 \\ \text{s.t. } \mathbf{D}_j \mathbf{D}_j^T \leq 1, \forall j = 1, \dots, l, \end{aligned}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^T \in R^{d \times n}$, \mathbf{x}_i is the d -dimensional feature vector, and n is the number of training samples. $\mathbf{D} \in R^{l \times d}$ is an overcomplete dictionary ($l > d$) with l prototypes. $\mathbf{C} \in R^{n \times l}$ corresponds to the sparse representation of \mathbf{X} . λ_1 is the regularization parameter, while \mathbf{D}_j denotes the j -th row of \mathbf{D} . The sparsity constraint prevents the learned dictionary from being arbitrarily large. Popular pooling strategies are average and max pooling. Average pooling is defined as $\mathbf{p} = \frac{1}{m} \sum_{i=1}^m \mathbf{c}_i$, while max pooling is defined as $p_k = \max\{|c_{1k}|, |c_{2k}|, \dots, |c_{mk}|\}$, where p_k is the k -th element of \mathbf{p} , c_{ij} is the element at position (i, j) in \mathbf{C} . m is the local number of local descriptors in the considered image region.

Experimental Results. Instead of pooling features from a regular spatial pyramid, Gilani et al. [2015] pooled sparse representations of scale-variant feature transform (SIFT) and color (CSIFT) features [Abdel-Hakim and Farag 2006] within a 30×30 window around fixated locations. Compared with conventional SIFT, CSIFT does not just embed color information in the descriptors, but also provides them robustness with respect to color variations as well as geometrical changes. Table 9.1 compares the impact of different pooling strategies on animal recognition over PET images using SIFT and CSIFT descriptors. All experiments were repeated five times and average accuracies and standard deviations reported. A linear SVM classifier was used as in Yang et al. [2009] and, as with their findings, for the baseline implementation that does not use fixation information, max pooling based on sparse codes generally outperforms average pooling on PET. When feature pooling is performed around fixated locations, max pooling achieves the best results on four animal classes with SIFT, and on three classes with CSIFT. A mean improvement of 3% in recognition performance is achieved via SIFT feature pooling around fixations with respect to max pooling over a regular spatial pyramid. A further 2% performance improvement is achieved with CSIFT, confirming that color is an important feature for recognizing animal objects. It is worth noting here that color is extremely useful for constructing both low-level visual features and high-level

Table 9.1 Object recognition accuracy with different pooling strategies on PET images (SIFT and CSIFT features)

SIFT	bird	cat	cow	dog	horse	sheep	avg+std
max pooling [Yang et al. 2009]	0.333 ± 0.018	0.263 ± 0.058	0.220 ± 0.032	0.522 ± 0.024	0.589 ± 0.036	0.437 ± 0.045	0.394 ± 0.007
avg pooling	0.323 ± 0.035	0.283 ± 0.026	0.222 ± 0.041	0.463 ± 0.018	0.517 ± 0.044	0.402 ± 0.051	0.368 ± 0.012
max pooling @ eye fixation (visual search)	0.357 ± 0.022	0.278 ± 0.034	0.253 ± 0.023	0.517 ± 0.016	0.659 ± 0.021	0.472 ± 0.031	0.423 ± 0.011
avg pooling @ eye fixation (visual search)	0.346 ± 0.021	0.291 ± 0.014	0.247 ± 0.036	0.508 ± 0.009	0.547 ± 0.022	0.441 ± 0.038	0.396 ± 0.021
max pooling @ eye fixation (free-viewing)	0.348 ± 0.019	0.264 ± 0.023	0.242 ± 0.019	0.499 ± 0.025	0.635 ± 0.018	0.457 ± 0.028	0.408 ± 0.009
avg pooling @ eye fixation (free-viewing)	0.341 ± 0.017	0.251 ± 0.018	0.224 ± 0.031	0.487 ± 0.012	0.526 ± 0.015	0.428 ± 0.026	0.376 ± 0.014
SIFT	bird	cat	cow	dog	horse	sheep	avg+std
max pooling [Yang et al. 2009]	0.357 ± 0.022	0.269 ± 0.032	0.231 ± 0.021	0.534 ± 0.021	0.592 ± 0.033	0.447 ± 0.032	0.412 ± 0.011
avg pooling	0.332 ± 0.031	0.299 ± 0.012	0.231 ± 0.021	0.481 ± 0.021	0.527 ± 0.034	0.422 ± 0.041	0.372 ± 0.021
max pooling @ eye fixation (visual search)	0.361 ± 0.023	0.288 ± 0.037	0.263 ± 0.031	0.527 ± 0.026	0.664 ± 0.031	0.484 ± 0.021	0.433 ± 0.018
avg pooling @ eye fixation (visual search)	0.365 ± 0.026	0.301 ± 0.017	0.252 ± 0.032	0.517 ± 0.011	0.562 ± 0.024	0.449 ± 0.021	0.419 ± 0.022
max pooling @ eye fixation (free-viewing)	0.346 ± 0.021	0.273 ± 0.014	0.252 ± 0.014	0.519 ± 0.023	0.655 ± 0.021	0.477 ± 0.031	0.421 ± 0.011
avg pooling @ eye fixation (free-viewing)	0.353 ± 0.021	0.262 ± 0.015	0.236 ± 0.028	0.499 ± 0.013	0.542 ± 0.013	0.447 ± 0.021	0.385 ± 0.013

saliency features. With fixation annotations incorporated, max pooling again outperforms average pooling for both SIFT and CSIFT. Finally, given the compactness of gaze patterns and the propensity of viewers to repeatedly fixate on target objects in the visual search task, 1–2% better recognition accuracy is achieved with eye fixations recorded during VS. This improvement can be attributed to cleaner target features available with semantically driven fixations, as compared to FV fixations that are more spread out and guided by bottom-up factors.

To summarize the results of the study, [Gilani et al. \[2015\]](#) compare eye movements acquired under the FV and VS conditions and noted the following. (1) While viewers are quicker at fixating on target animals and show greater urgency to scan the scene when instructed to perform visual search, the proportion of fixations on targets is similar for both conditions. (2) Objective measures computed via multi-match, recurrent quantification analysis, and contrast analysis confirm that viewer scan paths are more compact and semantically driven during visual search, while free-viewing trajectories are more exploratory and influenced by low-level scene properties. (3) In spite of the target animals receiving roughly equal attention in both conditions, the compactness of scan paths during VS allows for the learning of purer object signatures for recognition. Finally, enhanced animal recognition is noted when SIFT and CSIFT descriptors pooled from around the fixated locations are employed for learning target signatures. CSIFT outperforms SIFT, as it is more robust to geometrical changes. The following section describes how implicitly compiled physiological signals from users enable emotion and personality type recognition.

9.4 Emotion and Personality Type Recognition via Physiological Signals

The previous sections described how eye movements are useful for scene understanding and object recognition. Eye movements are also known to be indicators of emotional content. Before proceeding, it is important to define the categorical and dimensional models of emotion. The categorical model of emotions was proposed by [Ekman \[1992\]](#), and concludes that six basic emotions are universally recognizable, namely, anger, disgust, fear, happiness, sadness, and surprise. Nevertheless, psychologists prefer to work with dimensional models of emotion in practice, among which the circumplex emotion model proposed by [Russell \[1980\]](#) is most popular. This model proposes that emotions lie on a two-dimensional plane spanned by the valence and arousal dimensions. Valence characterizes the

perceived emotion as pleasant or unpleasant, while arousal describes emotional intensity. We will limit ourselves to the circumplex emotion model in this section.

Among other research that has examined the relationship between eye movements and emotions, [Subramanian et al. \[2014\]](#) investigate the influence of emotions on eye movements and memory for the gist and details of movie scenes. Their study observes systematic differences in eye movements for neutral and emotional clips, and concludes that emotional content attracts visual attention and strengthens memory for scene gist, while weakening memory for scene details. [R.-Tavakoli et al. \[2015\]](#) study the effectiveness of ten eye-movement based features for predicting scene valence. From their experiments, they conclude that fixation information and angular behavior of eye movements are the main valence indicators. In a recent multimedia study examining optimal points for inserting emotional advertisements within affective videos, [Yadati et al. \[2014\]](#) measure pupillary dilation as a proxy for induced arousal.

As mentioned earlier, implicit cues acquired from users can reveal information regarding the media they are interacting with, as well as the users themselves. This section describes a recent study by [Wache et al. \[2015\]](#), designed to simultaneously recognize (i) the emotion conveyed by movie clips and (ii) the big-five personality traits of users viewing the clips. The big-five or five-factor model [[Costa and McCrae 1992](#)] describes human personality in terms of five dimensions—extraversion (sociable vs. reserved), neuroticism or the degree of emotional stability (nervous vs. confident), agreeableness (compassionate vs. dispassionate), conscientiousness (dutiful vs. easy-going), and openness (curious/creative vs. cautious/conservative). A correlation between individuals' emotional behavior and personality traits was posited in H. J. Eysenck's personality model [1947]. Eysenck posited that (i) extraversion is accompanied by low cortical arousal (i.e., extraverts require more external stimulation than introverts), and (ii) neurotics become very easily upset or nervous due to minor stressors, while emotionally stable persons remain composed under pressure.

The influence of personality differences on users' affective behavior is examined in [Wache et al. \[2015\]](#), and emotion and personality recognition from physiological signals is attempted in this work.⁵ More specifically, user-expressed emotions and personality traits are characterized via heart rate, skin conductance, and brain (EEG) and facial activity patterns observed while viewing affective movie clips. This methodology of estimating personality traits from affective behavior is different

5. An extension of this work has been published in [Subramanian et al. \[2016\]](#).

from traditional methods that model personality traits based on (1) questionnaires or self-reports [Argamon et al. 2005], or (2) behavioral signals (typically non-verbal) acquired from social settings [Zen et al. 2010, Lepri et al. 2012, Subramanian et al. 2013, Alameda-Pineda et al. 2016b]. Furthermore, two aspects are unique to Wache et al. [2015]: (a) their study uses movie clips which are inherently intended to evoke emotions (and movie genres such as thriller, comedy, or horror are expressly defined by the emotions they evoke); and (b) they use commercial and wearable sensors for measuring physiological responses to ensure naturalistic user experience for ecological validity, and scalability for large-scale personality profiling. The details of this study are as follows.

9.4.1 Materials and Methods

An overview of the emotion and personality recognition framework employed by Wache et al. [2015] is presented in Figure 9.6 (top). To study the personality–affect relationship, physiological responses of 36 users in the form of the EEG brain signal and peripheral bio-signals (such as electrocardiogram (ECG) for measuring heart rate, galvanic skin response (GSR) for measuring skin conductance, and facial movements) were recorded as they viewed 36 affective movie clips used in Abadi et al. [2015]. Explicit feedback in the form of arousal, valence, liking, engagement, and familiarity ratings was obtained after subjects viewed each movie clip, but only arousal and valence ratings were analyzed in this study. Users’ personality measures for the big-five dimensions were compiled using a big-five marker scale (BFMS) questionnaire [Perugini and Di Blas 2002]. 36 university students (mean age = 29.2, 12 female) who were habitual Hollywood movie watchers participated in the study, and all subjects were fluent in English.

Materials. One PC with two monitors was used for the experiment. One monitor was used for video clip presentation at 1024×768 pixel resolution with 60 Hz screen refresh rate, and was placed roughly one meter in front of the user. The other monitor allowed the experimenter to check the recorded sensor data. Following informed consent, physiological sensors were positioned on the user’s body as shown in Figure 9.7(a). The GSR sensor (highlighted in green) was tied to the left wrist, and two electrodes were fixed to the index and middle finger phalanges. Two measuring electrodes for ECG (shown in red) were placed at each arm crook, with the reference electrode placed at the left foot. A single dry-electrode EEG device (marked in blue) was placed on the head like a normal headset, with the EEG sensor touching the forehead and the reference electrode clipped to the left ear. EEG data samples were logged using the Lucid Scribe software, and all sensor

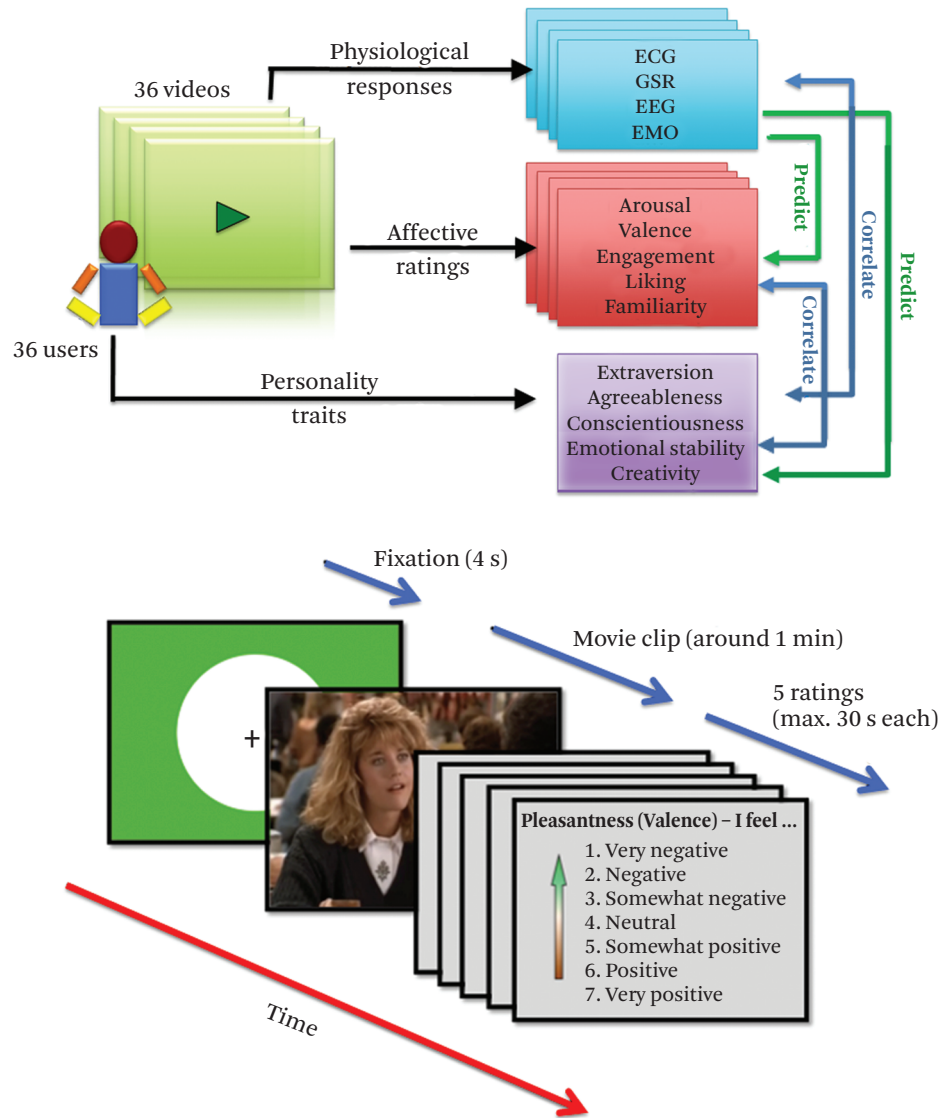


Figure 9.6 (top) Study overview. (bottom) Timeline for each trial. Best viewed under zoom. (Adapted from Wache et al. [2015])

data were recorded via Bluetooth. Also, a webcam was used to record facial activity. Synchronized data recording, pre-processing, and analyses were performed using MATLAB Psychtoolbox.⁶

Protocol. Each user performed the experiment in a session lasting about 90 minutes. Viewing of each movie clip was denoted as a trial. After two practice trials involving clips that were not part of the actual study, users watched movie clips randomly shown in two blocks of 18 trials, with a short break in-between to avoid fatigue. In each trial (Figure 9.6 (bottom)), a fixation cross was displayed for four seconds followed by clip presentation. On viewing each clip, users provided their affective ratings within a time limit of 30 seconds as described below. Participants also completed a personality questionnaire after the experiment.

Stimuli. 36 movie clips used in Abadi et al. [2015] were adopted for this study. These clips are between 51–127 sec long ($\mu = 80$, $\sigma = 20$), and are shown to be uniformly distributed (9 clips per quadrant) over the AV plane by Abadi et al. [2015].

Affective Ratings. For each movie clip, valence (V) and arousal (A) ratings were compiled reflecting the user's first impression. A 7-point scale was used with a –3 (very negative) to 3 (very positive) scale for V, and a 0 (very boring) to 6 (very exciting) scale for A. Ratings concerning engagement, liking, and familiarity were also acquired, but are not analyzed in this work. Mean user AV ratings for the 36 clips are plotted in Figure 9.7(b), and are color coded based on the ground-truth ratings provided in Abadi et al. [2015]. Affective ratings form a C-shape in the AV plane, consistent with prior studies [Koelstra et al. 2012, Abadi et al. 2015].

Personality Scores. Participants also completed the big-five marker scale (BFMS) questionnaire [Perugini and Di Blas 2002], which has been used in many personality recognition studies [Zen et al. 2010, Lepri et al. 2012, Subramanian et al. 2013]. Personality trait distributions along the big-five dimensions are shown in Figure 9.7(c).

9.4.2 Physiological Feature Extraction:

Affective physiological features corresponding to each trial over the final 50 seconds of stimulus presentation were extracted, owing to two reasons: (1) The clips used in Abadi et al. [2015] are not emotion-wise temporally homogeneous, but are more emotional toward the end. (2) Some employed features (see Table 9.2) are non-linear functions of the input signal length, and fixed time-intervals needed to be

6. <http://psychtoolbox.org/>

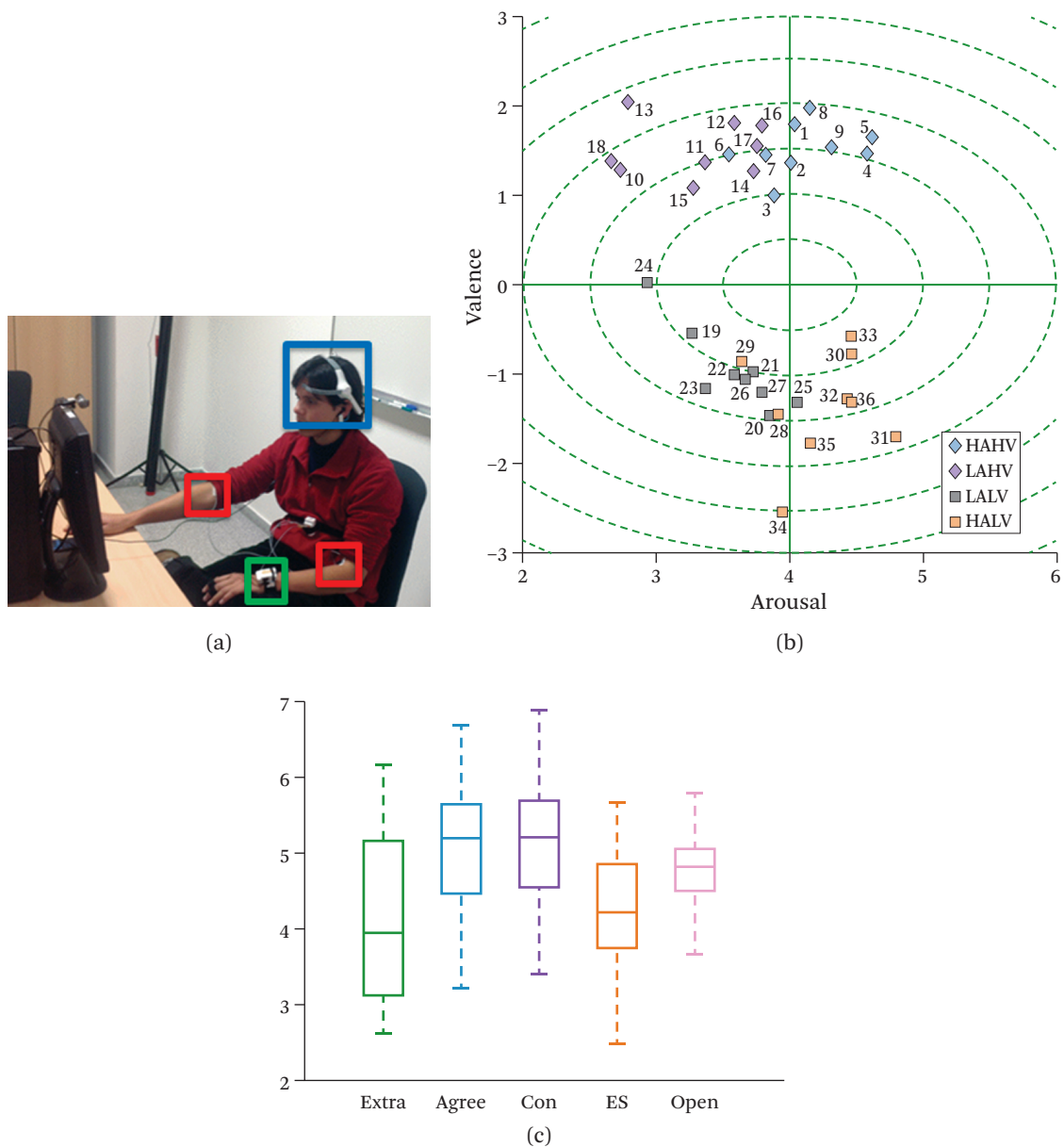


Figure 9.7 (a) Participant with sensors (EEG, ECG, and GSR sensors highlighted using blue, red and green boxes) during the experiment. (b) Mean arousal-valence (AV) ratings obtained for the 36 movie clips used. (c) Box-plots showing distribution of the big-five personality trait scores for 36 users. Figure best viewed under zoom. (Adapted from [Wache et al. \[2015\]](#))

Table 9.2 Extracted features for each modality

Modality	Extracted features
ECG (32)	Ten low-frequency ([0–2.4] Hz) power spectral densities (PSDs); four very slow response (VSR [0–0.04] Hz) PSDs; IBI, HR, and HRV statistics.
GSR (31)	Mean skin resistance and mean of derivative, mean differential for negative values only (mean decrease rate during decay time), proportion of negative derivative samples, number of local minima in the GSR signal, average rising time of the GSR signal, spectral power in the [0–2.4] Hz band, zero crossing rate of skin conductance slow response (SCSR) [0–0.2] Hz, zero crossing rate of skin conductance very slow response (SCVSR) [0–0.08] Hz, mean SCSR and SCVSR peak magnitude
Frontal EEG (88)	Average of first derivative, proportion of negative differential samples, mean number of peaks, mean derivative of the inverse channel signal, average number of peaks in the inverse signal, statistics over each of the 8 signal channels provided by the Neurosky software
EMO (72)	Statistics concerning horizontal and vertical movement of 12 motion units (MUs) specified in Joho et al. [2011]

Feature dimension stated in parentheses.

Statistics denote mean, standard deviation (std), skewness, kurtosis of the raw feature over time, and % of times the feature value is above/below $\text{mean} \pm \text{std}$.

considered as the movie clips were of varying lengths. A brief description of the extracted physiological features follows.

Galvanic Skin Response (GSR). GSR measures transpiration rate of the skin. When two electrodes are positioned on the middle and index fingers' phalanges and a small current is sent through the body, resistance to current flow changes with the skin transpiration rate. Most of the GSR information is contained in low-frequency components, and the signal is recorded at 100 Hz sampling frequency with a commercial and portable Bluetooth sensor. Following [Kim and Andre \[2008\]](#), [Koelstra et al. \[2012\]](#), and [Soleymani et al. \[2012\]](#), 31 GSR features are extracted as listed in [Table 9.2](#).

Electroencephalography (EEG). EEG measures changes in the skull's electrical field produced by brain activities, and information is encoded in the EEG signal ampli-

Table 9.3 Mean correlations between personality scales and user V and A ratings for all movie clips

	Ex	Ag	Con	ES	Open
Arousal	-0.15 ^a	-0.05	-0.04	0.07	-0.08 ^a
Valence	0.22 ^a	0.16	-0.06	0.08	0.15

a. Denotes significant correlations ($p < 0.01$) obtained using Fisher's method.

tude as well as in certain frequency components. A commercial, single dry-electrode EEG sensor⁷ is used in this work, which records eight information channels sampled at 32 Hz. The recorded information includes frontal lobe activity, level of facial activation, and eye-blink rate and strength, which are relevant emotional responses.

Electrocardiogram (ECG). Heart rate characteristics have been routinely used for user-centered emotion recognition. R-peak detection on the ECG signal is performed to compute users' inter-beat intervals (IBI), heart rate (HR), and the heart rate variability (HRV). Also, power spectral density (PSD) is extracted in low frequency bands as in [Kim and Andre \[2008\]](#) and [Soleymani et al. \[2012\]](#).

Facial landmark trajectories (EMO). A facial feature tracker [[Joho et al. 2011](#)] is used to compute displacements of 12 interest points or motion units in each video frame. Six statistical measures for each landmark are computed to obtain a total of 72 features (Table 9.2).

9.4.3 Personality Scores vs. Affective Ratings

From the personality scores and affective ratings compiled in the user study, the authors examined relationships between valence (V) and arousal (A) ratings and personality trait scores in the context of hypotheses put forth in literature. (i) Correlations between users' personality scales and V and A ratings were computed for each movie clip, and significant correlations were determined according to Fisher's method (Table 9.3); (ii) Personality measures were dichotomized based on the median score for each dimension to determine high/low trait groups (e.g., extraverts and introverts), and the affective ratings of each group were studied. Details of the analyses are as follows.

7. www.neurosky.com

Extraversion vs. Arousal and Valence. The correlation between extraversion (Ex) and arousal (A) has been investigated in many studies: EEG measurements in [Stenberg \[1992\]](#), signal detection analysis in [Gupta and Nicholson \[1985\]](#), and fMRI analysis in [Kehoe et al. \[2012\]](#) have shown lower arousal in extraverts as compared to introverts, consistent with Eysenck's theory. Also, Ex has been found to correlate with positive valence (V) in a number of studies [[Costa and McCrae 1980](#)]. Correlation analyses presented in [Table 9.3](#) confirm a slight but significant negative correlation between Ex and A as noted previously. Likewise, a significant positive correlation is noted between Ex and V mirroring prior findings. Therefore, previous observations connecting Ex and affective behavior are evidenced by the data.

Neuroticism vs. Arousal. The relationship between Neuroticism (Neu) and A has been extensively studied and commented on; a positive correlation between Neu and A is revealed through fMRI responses [[Kehoe et al. 2012](#)], and EEG analysis [[Stenberg 1992](#)] reinforces this observation especially for negative valence stimuli. [Ng \[2009\]](#) further remarks that neurotics experience negative emotions more strongly than emotionally stable persons. As correlation observed between Neu scores and A ratings in [Table 9.3](#) is not significant, *t*-tests were used to compare mean A ratings provided by the neurotic and emotionally stable (ES) groups upon dichotomization of the Neu scale via the median score (which was midway between extremes, resulting in equal-sized groups). To examine if the data suggested a positive correlation between Neu and A, authors performed a left-tailed *t*-test comparing the A ratings of the ES and neurotic groups. The test revealed a significant difference in A ratings for high A clips ($t(34) = -1.8058, p = 0.0399$), and a marginally significant difference for low A clips ($t(34) = -1.4041, p = 0.0847$). Quadrant-wise distributions of A ratings for the ES and neurotic groups are presented in [Figure 9.8\(a\)](#). Quadrant-wise comparisons show that neurotics generally experience higher A than ES subjects. Left-tailed *t*-tests confirm the significantly higher A ratings provided by neurotics for HAHV ($t(16) = -2.5828, p < 0.0100$) clips, and marginally higher A ratings for LALV ($t(16) = -1.6077, p = 0.0637$) and HALV ($t(16) = -1.3859, p = 0.0924$) clips. No difference however was observed for LAHV clips ($t(16) = -0.9946, n.s.$). In general, the analyses reveal that Neu is associated with higher A.

Neuroticism vs. Valence. Differing observations have been made regarding the relations between Neu and V. A negative correlation between Neu and positive V is observed in [Kehoe et al. \[2012\]](#), while a positive relationship between the two for low A stimuli is noted in [Tok et al. \[2010\]](#). [Ng \[2009\]](#) remarks that the Neu-V relation

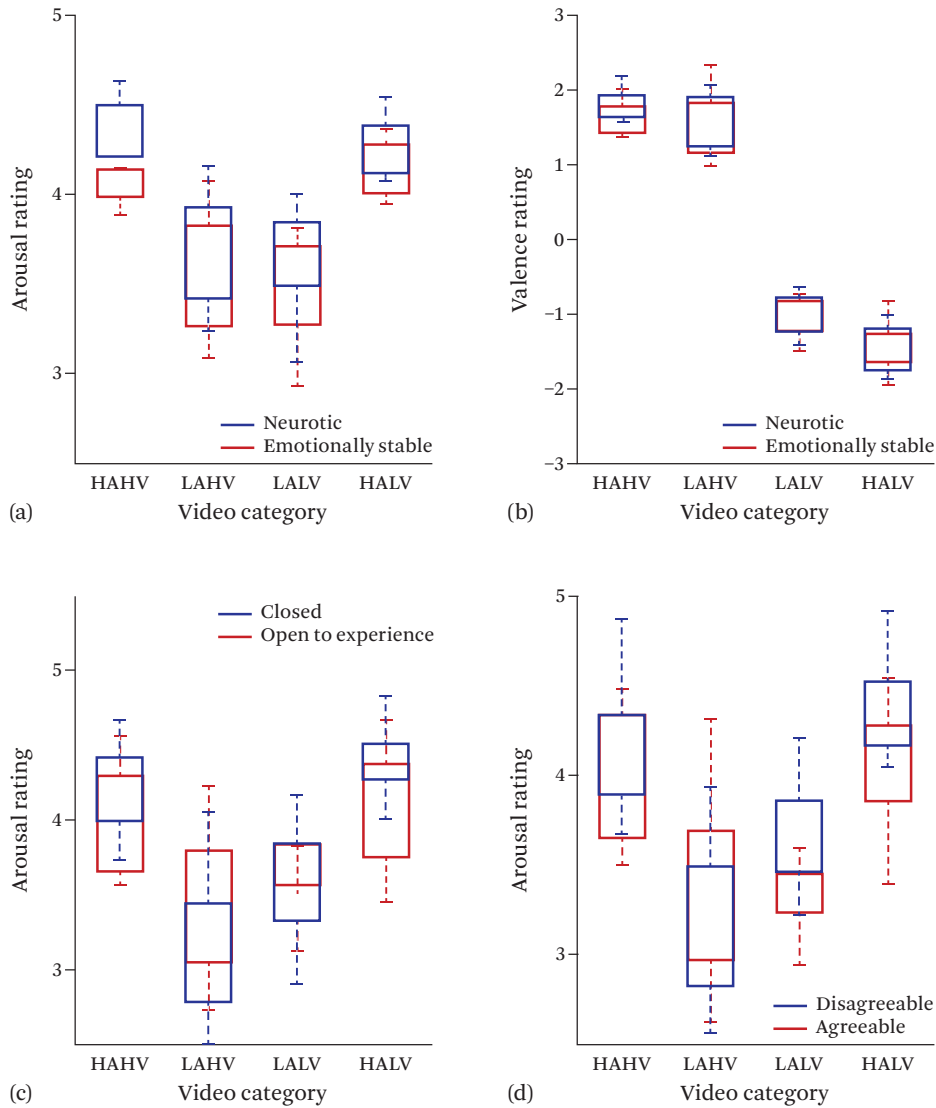


Figure 9.8 Quadrant-wise comparisons of A and V ratings for neurotics vs. emotionally stable subjects are shown in (a,b). Arousal rating comparisons for (c) open vs. closed, and (d) agreeable vs. disagreeable subjects. (Adapted from Wache et al. [2015])

is moderated by situation— while neurotics may feel less positive in unpleasant situations, they experience positive emotions as strongly as ES subjects in pleasant conditions. As no significant correlation between Neu and V is noted in Table 9.3, V ratings of the neurotic and ES groups are examined. Very similar V ratings are noted for high/low valence clips. Quadrant-wise V rating comparisons (Figure 9.8(b)) reveal that neurotics feel slightly more positive than ES subjects on viewing HAHV clips but the difference is not significant ($t(16) = -1.489, p = 0.1558$). Overall, analyses do not reveal any definitive relationship between Neu and V.

Openness vs. Arousal and Valence. Tok et al. [2010] note a positive correlation between Openness (Op) and V under low arousal conditions, which is attributed to the intelligence and sensitivity of creative individuals,⁸ enabling them to better appreciate subtly emotional stimuli. To examine this hypothesis, the authors used right-tailed *t*-tests comparing V and A ratings of the groups open and closed to experience upon dichotomization. Very similar V ratings were noted for the open and closed groups. Quadrant-based comparisons revealed that open individuals experienced only slightly higher V while viewing low arousal, high valence (LAHV) clips ($t(16) = 1.4706, p = 0.0804$). A significant and slightly negative correlation between Op and A can be noted from Table 9.3. Focusing on A ratings of open vs. closed groups, marginally different A ratings ($t(34) = -1.5767, p = 0.0621$) are noted for high arousal stimuli. Left-tailed *t*-tests for quadrant-based comparisons (Figure 9.8(c)) revealed that closed individuals experienced significantly higher A for high arousal, low valence (HALV) clips ($t(16) = -1.9834, p = 0.0324$), and slightly higher A for high arousal, high valence (HAHV) clips ($t(16) = -1.5402, p = 0.0715$). In summary, there is a negative relationship between Op and A, and a slightly positive relationship between Op and V as noted in Tok et al. [2010].

Agreeableness and Conscientiousness. Comparison of A ratings of the agreeable and disagreeable groups (Figure 9.8(d)) revealed that agreeable individuals are generally less aroused by LV videos ($t(34) = -2.1859, p = 0.0358$). Quadrant-based analyses also show that disagreeable individuals felt more aroused by HALV ($t(16) = -2.5493, p = 0.0214$), and marginally more aroused low arousal, low valence (LALV) ($t(16) = -2.0976, p = 0.0522$) clips. Conscientiousness scale differences did not significantly influence the V and A ratings in any manner.

8. Creativity strongly correlates with openness [Mervielde et al. 1998].

Table 9.4 Affective state recognition from physiological signals with linear SVM and naive Bayes (NB) classifiers

	ECG		GSR		EMO		EEG		Peripheral		W _{est} ^t		Class Ratio
	SVM	NB	SVM	NB	SVM	NB	SVM	NB	SVM	NB	SVM	NB	
Arousal	0.54	0.58	0.52	0.58	0.55	0.60	0.59	0.61	0.56	0.62	0.62	0.62	0.50
Valence	0.55	0.58	0.52	0.57	0.58	0.62	0.60	0.62	0.55	0.62	0.64	0.64	0.50

Mean F1-scores over all participants for the four modalities, peripheral signals (ECG + GSR), and late fusion (W_{est}^t) are shown.

Baseline F1-score is 0.5. Maximum unimodal F1-scores are shown in bold.

9.4.4 Emotion and Personality Recognition from Physiological Signals

In the previous section, a direct correlation with affective ratings was noted only for the Ex and Op traits. For the other traits, the influence of personality differences on users' affective behavior was revealed only via quadrant-wise comparisons, where affective ratings of the high and low trait groups for emotionally similar clips were examined. While affective ratings represent an explicit or conscious reflection of one's emotional state, it would be reasonable to expect physiological responses to implicitly convey the same information, and thereby reveal personality differences. [Wache et al. \[2015\]](#) attempt emotion and personality recognition from physiological features, and these results are tabulated in [Table 9.4](#) and [Table 9.5](#).

Emotion and Personality Trait Recognition. As the work of [Wache et al. \[2015\]](#) exclusively uses commercial sensors for examining users' physiological behavior, the authors follow a procedure identical to DEAP [[Koelstra et al. 2012](#)] to benchmark their results with prior affective studies employing lab-grade sensors. To this end, the most discriminative physiological features are first identified for each modality using Fisher's linear discriminant with a threshold of 0.3. Features corresponding to each user are then fed to the naive Bayes (NB) and linear SVM classifiers as shown in [Table 9.4](#). A leave-one-out cross-validation scheme is employed, where one video is held out for testing, while the other videos are used for training. The best misclassification cost parameter C for linear SVM is determined via grid search over $[10^{-3}, 10^3]$, again using leave-one-out cross-validation.

[Table 9.4](#) presents the mean F1-scores over all users, obtained using the NB and SVM classifiers with unimodal features and the decision fusion (W_{est}^t) technique described in [Koelstra and Patras \[2013\]](#). In decision fusion, the test sample label is computed as $\sum_{i=1}^4 \alpha_i^* t_i p_i$. Here, i indexes the four modalities used in this work, p_i 's denote posterior SVM probabilities, $\{\alpha_i^*\}$ are the optimal weights maximizing the

Table 9.5 Personality trait recognition considering affective responses to (a) all, and (b) emotionally homogeneous stimuli

Videos	Method	Extravert		Agreeable		Conscient		Em. Stab.		Open	
		acc	F1	acc	F1	acc	F1	acc	F1	acc	F1
All	ECG	0.45	0.43	0.42	0.37	0.45	0.31	0.52	0.50	0.55	0.54
	GSR	0.15	0.14	0.42	0.34	0.39	0.28	0.24	0.20	0.91	0.91
	EMO	0.61	0.59	0.42	0.34	0.12	0.11	0.42	0.37	0.27	0.26
	EEG	0.36	0.34	0.39	0.28	0.15	0.13	0.55	0.50	0.55	0.50
	W _{est} ^t	0.61	0.60	0.42	0.39	0.45	0.31	0.64	0.63	0.91	0.91
HAHV	ECG	0.69	0.69	0.75	0.75	0.19	0.18	0.63	0.63	0.06	0.06
	GSR	0.59	0.59	0.78	0.78	0.25	0.25	0.31	0.29	0.78	0.78
	EMO	0.31	0.31	0.84	0.84	0.41	0.41	0.47	0.44	0.22	0.21
	EEG	0.09	0.09	0.72	0.72	0.34	0.34	0.41	0.39	0.53	0.53
	W _{est} ^t	0.78	0.78	0.84	0.84	0.56	0.56	0.69	0.69	0.78	0.78
HALV	ECG	0.45	0.45	0.76	0.76	0.34	0.26	0.45	0.41	0.55	0.55
	GSR	0.72	0.72	0.72	0.72	0.55	0.55	0.21	0.19	0.69	0.69
	EMO	0.38	0.34	0.38	0.37	0.38	0.34	0.52	0.51	0.62	0.62
	EEG	0.34	0.32	0.24	0.23	0.69	0.69	0.31	0.29	0.62	0.61
	W _{est} ^t	0.72	0.72	0.79	0.79	0.76	0.76	0.55	0.54	0.69	0.69
LAHV	ECG	0.45	0.39	0.32	0.31	0.42	0.34	0.55	0.45	0.58	0.56
	GSR	0.32	0.27	0.45	0.44	0.42	0.34	0.42	0.30	0.77	0.77
	EMO	0.65	0.63	0.26	0.23	0.65	0.64	0.65	0.63	0.29	0.25
	EEG	0.32	0.30	0.65	0.63	0.65	0.62	0.52	0.52	0.68	0.67
	W _{est} ^t	0.68	0.67	0.68	0.67	0.65	0.64	0.74	0.74	0.81	0.81
LALV	ECG	0.50	0.49	0.20	0.20	0.13	0.13	0.37	0.27	0.30	0.29
	GSR	0.27	0.26	0.23	0.23	0.33	0.33	0.60	0.57	0.63	0.63
	EMO	0.43	0.43	0.47	0.46	0.27	0.27	0.30	0.26	0.17	0.16
	EEG	0.33	0.33	0.00	0.00	0.10	0.10	0.67	0.66	0.37	0.35
	W _{est} ^t	0.57	0.54	0.53	0.53	0.33	0.33	0.70	0.69	0.63	0.63

HAHV = High Arousal, High Valence; HALV = High Arousal, Low Valence; LAHV = Low Arousal, High Valence; LALV = Low Arousal, Low Valence.

Maximum F1-scores with unimodal and multimodal methods are shown in bold.

F1-score on the training set, and $t_i = \alpha_i F_i / \sum_{i=1}^4 \alpha_i F_i$, where F_i denotes the F1-score obtained on the training set with the i^{th} modality. An equal number of high/low A and V clips are used in the study, implying a class ratio (and consequently, a baseline F1-score) of 0.5.

Observing Table 9.4, above-chance ER is evidently achieved with physiological features extracted using commercial sensors. The obtained F1-scores are very similar to DEAP [Koelstra et al. 2012], which can possibly be attributed to the use of movie clips that are found to be optimal for emotional inducement as discussed in Abadi et al. [2015]. EEG features produce the best recognition performance for both A and V, and both EEG and facial features produce best recognition for V. GSR produces the worst recognition performance, and the NB classifier outperforms linear SVM for all considered features. The best fusion-based recognition performance of 0.64 is noted for V, and better (unimodal as well as multimodal) recognition is generally noted for V as in Koelstra et al. [2012] and Abadi et al. [2015].

Recognition accuracies and F1-scores achieved using the above-computed physiological features for the five personality dimensions are presented in Table 9.5. Upon dichotomizing the personality scores based on the median, an inexact split (19 vs. 17) was obtained only for the conscientiousness (Con) and Op traits. Therefore, baseline accuracy/F1-score for these two traits is 0.53, while being 0.5 for the others. Considering affective responses to all videos, better-than-chance recognition is achieved only for three traits (excepting Ag and Con). When physiological responses to emotionally similar videos are considered, the best unimodal F1-scores for all personality traits are above-chance excepting Con with HAHV clips. These results confirm that recognition of high and low traits is more easily achieved by comparing users' affective responses to emotionally similar clips. Consistently high recognition performance is achieved for Op, and also for Ag considering homogeneous videos. Con is the most difficult trait to recognize using the considered physiological features. Focusing on the sensing modalities, GSR consistently produces the best performance for Op while EMO performs best overall (highest F1-score in 8/25 conditions).

Discussion of Experimental Results. The study of Wache et al. [2015] suggests that both emotions and personality differences can be recognized from user physiological responses captured with commercial and wearable sensors. Also, personality differences are captured better via physiological features when user responses to emotionally homogeneous videos are considered. Better-than-chance recognition of emotions is noted, while considerably better-than-chance recognition is achieved with either unimodal or multimodal features for all personality traits

considering homogeneous clips (excepting Con with LALV clips). Best recognition results are obtained for Op and Ag, while worst performance is noted for Con. It is pertinent to point out some limitations of the study in general. Apart from the small sample size of users examined in the study, weak linear correlations are noted between emotional and personality attributes in Table 9.3 implying that the personality–affect relationship may not be optimally captured via linear correlations. Overall, promising emotion and personality trait recognition is achieved with the proposed framework employing minimally intrusive, wearable, and commercial sensors.

9.5 Conclusion

In this chapter, we have reviewed some studies that employ implicit user cues such as eye movements and bio-signals for inference about media content as well as the users themselves. We will conclude this chapter by providing some insights regarding how implicit user cues can be utilized to solve the research problems of today and tomorrow.

Safety and Security. The emphasis on public safety and security has markedly increased over the past decade. As driver fatigue has been deemed to account for 40% of road accidents, considerable effort has recently focused on autonomous cars, even though the technology is still in its infancy and faces a lot of challenges. While it may be difficult to entirely relieve the driver from the burden of driving, a situation where the autonomous driving control temporarily takes over from a fatigued driver is very conceivable. Likewise, assessing the fatigue level of security officers who perform repetitive yet critical surveillance tasks through the day is an important research problem. A number of recent studies have focused on fatigue detection by monitoring eye movements [Abdulin and Komogortsev 2015] and cognitive behavior [Zhang et al. 2015b]. As fatigue is a multi-dimensional problem that can result in an inability to maintain optimal muscular performance or in an inhibition of cognitive capabilities, challenges in this field involve the design of a holistic and multimodal framework that can efficiently process user bio-signals for real-time fatigue detection and measurement.

Interaction Design. Over the last decade, there has been a conscious move to consider cognitive and emotional aspects of user experience (UX) alongside standard usability requirements, and implicit user behavioral signals provide a handle to users' mental and emotional state as seen in this chapter. Nevertheless, the emphasis of late has been on designing inexpensive, minimally intrusive and ecologically

valid systems such as wearable devices; the UX-mate proposed by [Staiano et al. \[2012\]](#) is a prototypical example in this regard. Study of eye movements has greatly contributed to webpage design, and discovery of the “F-Pattern” as users visually scan webpages⁹ has facilitated the formulation of effective design guidelines. Also, implicit behavioral signals can help in the objective compilation of real-time design feedback from users, traditionally achieved using questionnaires which (1) only reflect aggregated opinions on holistic user experience, and (2) require additional user time and effort post-task completion.

Big-data Visualization. With the explosion of big-data in today’s world, the need for information visualization or InfoVis systems for exploratory data analysis and decision-making has gained importance. Design and usability heuristics used to evaluate traditional interfaces are unsuitable for InfoVis systems, as providing insights (e.g., discovering graph patterns) is one of their key functions apart from facilitating a variety of user interactions. In this respect, the ability to objectively monitor the user’s cognitive state and in particular measuring the cognitive load on the interacting user via the EEG and fNIRS (functional near-infrared spectroscopy) modalities has gained traction recently (see [Solovey et al. \[2012\]](#) and [Peck et al. \[2013\]](#) for recent related work). Challenges in this domain include real-time bio-signal processing, and precise mapping of the monitored signal properties to cognitive states.

Overall, while user behavioral (bio) signals carry rich information, they are nevertheless noisy, which calls for the development of robust signal processing and machine learning techniques for efficiently harnessing them. If the accompanying challenges can be addressed, behavioral signals can be effectively used for solving many topical and open research problems via human-machine synergy.

9. <http://thenextweb.com/dd/2015/04/10/how-to-design-websites-that-mirror-how-our-eyes-work/>



PART

**MULTIMEDIA
COMMUNICATION
AND SYSTEMS**

Multimedia Fog Computing: Minions in the Cloud and Crowd

Cheng-Hsin Hsu (National Tsing Hua University),
Hua-Jun Hong (National Tsing Hua University),
Tarek Elgamal (University of Illinois, Urbana-Champaign),
Klara Nahrstedt (University of Illinois,
Urbana-Champaign),
Nalini Venkatasubramanian (University of California,
Irvine)

In cloud computing, minions refer to virtual or physical machines that carry out the actual workload. Minions in the cloud hide in faraway data centers and thus cloud computing is less friendly to multimedia applications. The fog computing paradigm pushes minions toward edge networks. We adopt a generalized definition, where minions get into end devices owned by the crowd. The serious uncertainty, such as dynamic network conditions, limited battery levels, and unpredictable minion availability in multimedia fog platforms makes them harder to be managed than cloud platforms. In this chapter, we share our experience on utilizing resources from the crowd to optimize multimedia applications. The learned lessons shed some light on the optimal design of a unified multimedia fog platform for distributed multimedia applications.

10.1 Introduction

Cloud computing has become mature in the past decade, and enabled many new applications, such as remote collaboration, file backup, online office suites, and

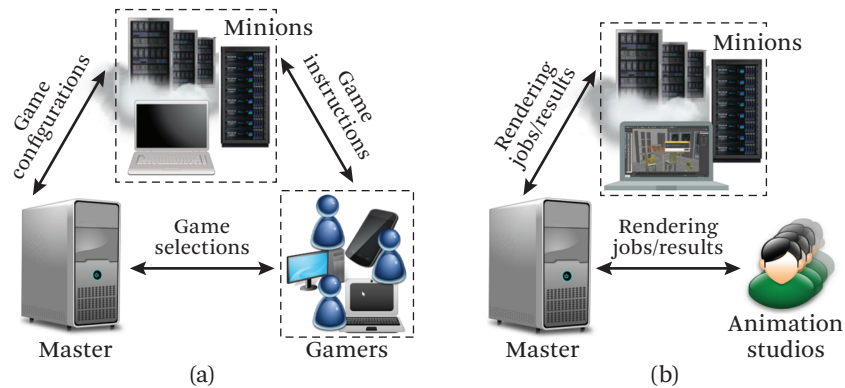


Figure 10.1 Sample multimedia applications: (a) cloud gaming and (b) animation rendering.

cloud gaming. These cloud applications are executed in a distributed manner, and dictate efficient management stacks, such as OpenStack (<http://www.openstack.org>), SaltStack (<http://saltstack.com>), and Kubernetes (<http://kubernetes.io>), in which a master oversees the operations, while one or multiple minions carry the actual workload. The master and minions may be deployed on physical machines or in virtual machines (VMs). Conventional cloud computing deploys masters and minions in data centers, for elasticity, flexibility, reliability, security, and cost-effectiveness. Such a deployment approach, however, is not suitable to distributed multimedia applications, which simultaneously require: (i) stringent realtime constraints and (ii) tremendous amounts of resources. For example, to achieve acceptable user experience, cloud gaming services, shown in Figure 10.1(a) have to finish the following jobs in as short as 100 ms [Claypool and Claypool 2006]: (i) receiving and processing client inputs on minions serving as game servers; (ii) encoding and transmitting the captured video frames from game servers to gamers' clients; and (iii) receiving, decoding, and rendering video frames on clients' displays. Another example is animation rendering, shown in Figure 10.1(b) which needs large amounts of resources, such as: (i) high-speed CPU/GPU for rendering, (ii) large disk space for storage, and (iii) fast networks for data transfers. Hosting these multimedia applications in the cloud is not ideal for service quality, because minions hiding in data centers are too far away from users and are vulnerable to insufficient bandwidth, long latency, and network outage. Furthermore, cloud providers often ask for higher rates on heavier (multimedia) users, making these multimedia applications less commercially viable when being hosted in the cloud.

Fog computing was recently proposed by the networking community [Bonomi et al. 2012], for Internet-of-Things (IoT) applications. Fog computing extends cloud

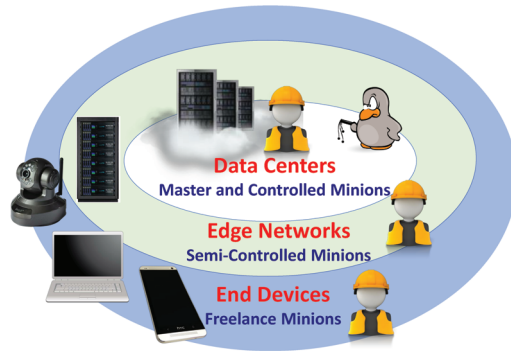


Figure 10.2 Overview of a fog computing platform.

computing from data centers toward the edge networks, so as to achieve: (i) low latency, (ii) location awareness, (iii) scalability, and (iv) heterogeneity, which are crucial to multimedia applications. That is, minions can now be hosted on less-powerful networking devices or dedicated machines in edge networks for better realtimeness. The definition of fog computing was further generalized [Vaquero and Merino 2014, Yi et al. 2015], where minions are pushed to heterogeneous end devices owned by general publics. That is, end users, or the crowds, are somehow incentivized to share their storage and processing resources to run minions for basic network functionalities or novel applications. Outsourcing minion workload to general publics is one kind of crowdsourcing [Yuen et al. 2011], which has a potential for fog service providers to sell, otherwise idling resources at lower costs. In the resulting fog computing platform, we refer to: (i) the minions in data centers as *controlled* minions, (ii) the minions in edge networks as *semi-controlled* minions, and (iii) the minions on end devices as *freelance* minions, as summarized in Figure 10.2.

We envision a general multimedia computing platform, called the *multimedia fog platform*, in which minions (fog devices) in data centers, edge networks, and end devices leverage four resources: *communications*, *computations*, *storage*, and *sensors*, for next-generation distributed multimedia applications. As illustrated in Figure 10.3, the master leverages diverse resources from minions to accomplish the jobs/requests from fog users. Managing minions in the multimedia fog platform is harder than doing that in the cloud platform for various reasons: for example, end devices may move into wireless dead-zones, run out of battery, and be turned off anytime, leading to serious uncertainty. In fact, a wide spectrum of challenges must be addressed before realizing distributed multimedia applications on the multimedia fog platform, including resource discovery, network Quality-of-Service

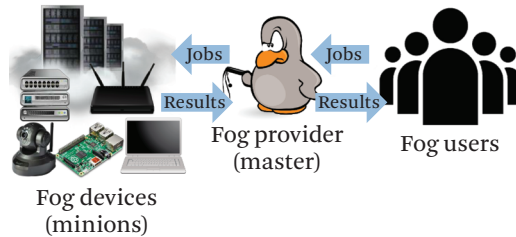


Figure 10.3 Interactions among the fog provider (master), fog devices (minions), and fog users.

(QoS), user Quality-of-Experience (QoE), network/device heterogeneity, resource allocation, management, security, programmability, and accountability. That is, there exist a huge room for optimizing the multimedia fog platform.

10.2 Related Work

Several studies in the literature can be seen as special cases of our multimedia fog platform. While the research problems in our multimedia fog platform may be more challenging, the experience gained in related work still sheds some light on potential solutions. These studies are classified into the following groups, which are briefly surveyed.

Volunteer Computing. In a 2001 paper, [Sarmenta](#) proposes the volunteer computing concept to aggregately utilize the unused resources of desktops for computationally demanding jobs. SETI@Home [[Anderson et al. 2002](#)] is a volunteer computing application that analyzes radio signals from the space for extraterrestrial intelligence. BOINC [[Anderson 2004](#)] is a generalized platform of volunteer computing, which utilizes computers connected to the Internet for diverse computational jobs. Unlike with our multimedia fog platform, the volunteer computing workers are attracted by some high profile projects, such as finding aliens, analyzing protein structures, and discovering antimalarial drugs for free. Hence, volunteer computing does not suffer from some challenges of the multimedia fog platforms. For example, fog providers need to carefully manage resources consumed by each job to remain profitable. In addition, fog providers must guarantee that each job is completed in time in order to retain fog users.

Mobile Offloading. To solve the problem of running resource-hungry jobs on mobile devices, researchers propose to offload these jobs to powerful cloud servers. [Lin et al. \[2013\]](#) propose a context-aware mobile cloud offloading decision engine,

which considers several contexts, such as signal strength and GPS locations, to make the offloading decisions. The decisions may reduce energy consumption or response time. Researchers also offload the resource-hungry jobs to edge networks, e.g., [Satyanarayanan et al. \[2009\]](#) propose Cloudlet, which is a server placed next to a WiFi access point. Mobile devices connect to Cloudlet servers and offload their jobs through one-hop wireless networks. [Willis et al. \[2014\]](#) propose ParaDrop, which allows mobile users to run their jobs on WiFi access points. Researchers also try to offload the jobs among multiple mobile devices. For example, [Verbelen et al. \[2012\]](#) extend the Cloudlet by aggregating nearby mobile devices for computationally demanding jobs. In contrast to mobile offloading, our multimedia fog platform uses heterogeneous devices and supports diverse applications, including computing-, communication-, storage-, and sensing-intensive applications.

P2P Cloud. The P2P (peer-to-peer) paradigm has been adopted to solve various problems, including video streaming and file sharing. For example, [Liu et al. \[2008\]](#) survey the P2P video streaming systems and [Pouwelse et al. \[2005\]](#) conduct a measurement study on BitTorrent. In general, P2P systems are more scalable compared to client-server systems. Researchers also propose fully distributed P2P clouds in the literature, e.g., [Xu et al. \[2009\]](#) study P2P cloud storage. [Babaoglu et al. \[2012\]](#) design and implement a P2P cloud system, and propose an algorithm to partition the resources for better performance. [Graffi et al. \[2010\]](#) propose a protocol for resource reservations, such as storage space and network bandwidth. Unlike our multimedia fog platform, P2P clouds are fully distributed systems without any central entity and thus face challenges to: (i) manage end devices owned by crowds and (ii) allocate the resources to provide QoS guarantees.

Cloud Computing. Cloud computing is designed for providing on-demand resources with remote powerful data centers. Several studies [[Weinhardt et al. 2009](#), [Chang et al. 2010](#), [Dillon et al. 2010](#)] investigate the business models of cloud computing. These studies present a business model with a multi-level cloud [[Weinhardt et al. 2009](#)], design cost and charging models [[Chang et al. 2010](#)], and analyze pros/cons of different business models [[Dillon et al. 2010](#)]. Several other papers [[Hong et al. 2015](#), [Beloglazov et al. 2012](#), [Mishra et al. 2012](#), [Gong et al. 2010](#)] study the resource management problem. For example, they design the resource allocation algorithms to optimize energy consumption [[Beloglazov et al. 2012](#)] and profits [[Hong et al. 2015](#)]. Moreover, [Mishra et al. \[2012\]](#) leverage live migration for dynamic resource allocation, and [Gong et al. \[2010\]](#) utilize a prediction algorithm to optimize resource allocation of future workload. Some studies [[Buyya et al. 2011](#),

[Wu et al. 2011, Van et al. 2009] tackle SLA- (service-level-agreement) and QoS-driven resource allocation problems to guarantee the SLA/QoS. Patel et al. [2009] propose an architecture for monitoring, managing, and evaluating the QoS/SLA guarantees. Alhamad et al. [2010] propose a trust model to convince customers that the resulting QoS/SLA guarantees are met. Exposing data and services in public networks leads to security and privacy concerns, such as intrusion, data integrity, and auditing issues [Zhou et al. 2010]. Li et al. [2010] design an efficient searching algorithm for encrypted data. Shacham and Waters [2008] ensure data integrity. Wang et al. [2010, 2013] propose a third-party auditing concept. Unlike in our multimedia fog platform, the devices used in the cloud are all fully controlled, and their heterogeneity is rather low.

10.3 Challenges

Designing a general multimedia fog platform leads to many challenges because of the uncertainty of the semi-controlled and freelance minions. Figure 10.4 summarizes the major challenges, which are detailed below.

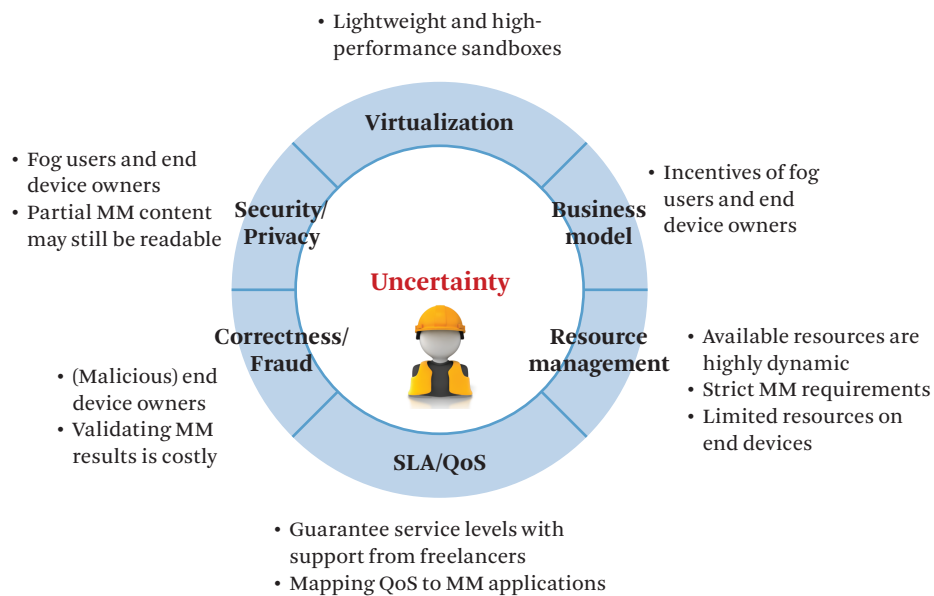


Figure 10.4 Challenges of developing a general multimedia fog platform.

Business Model. A well-designed business model is crucial to any commercial systems, including our multimedia fog platform. Creating one is quite difficult because fog providers have to consider fog users, semi-controlled minions, and freelance minions at the same time. For example, fog providers need to attract fog users from clouds, and freelance minions need to be incentivized to contribute their idling resources. On top of that, fog providers must make profits to keep their services sustainable. Several non-monetary incentive mechanisms [Deterding et al. 2011, Anderson 2004] may be considered in our multimedia fog platform for higher profit margins. In summary, attracting the fog users, motivating the end device contributors, and earning enough money (for fog providers) at the same time are essential to a successful multimedia fog platform.

Virtualization. Virtualization has been adopted in various applications Satyanarayanan et al. [2009] propose to offload resource-hungry mobile applications to nearby resourceful Cloudlet servers running VMs. Compared to dedicated Cloudlet servers managed by companies, the multimedia fog platform is much more dynamic. For example, the owner of a freelance minion may play games in the morning and have more available resources in the evening. To achieve high quality and profits, the multimedia fog platform needs to frequently migrate the services across servers and end devices to adapt to the system dynamics, which demand lightweight virtualization technologies. Choosing the best lightweight virtualization technology is a challenging problem. There is an emerging lightweight virtualization technology called containers, such as LXC (<http://linuxcontainers.org>) and Docker (<http://www.docker.com>). Containers seem to be promising, but there are still quite a few open issues, e.g., running containers on Android or iOS devices is not possible now.

Resource Management. Fog providers have to design efficient resource allocation algorithms to manage the resources across multiple minions. Several objectives are possible, including maximizing the fog service quality at a given target profit level, or maximizing the profit at a minimum service quality level. While similar resource management has been done on cloud platforms in data centers, carrying out performance measurement, modeling, and optimization in multimedia fog platforms is much harder. This is because the resources provided by the semi-controlled and freelance minions are not as powerful as servers in data centers. Some complicated multimedia applications cannot be served by a single semi-controlled or freelance minion, e.g., running an object recognition application on an embedded system in realtime may be infeasible. Fog providers have to decompose the applications

to smaller components and assign these components to the right minions that have enough resources. Moreover, semi-controlled and freelance minions are dynamic and cannot be fully controlled. It is hard to predict the amount of available resources that can be allocated to multimedia applications during future time slots.

SLA/QoS. SLA guarantees refer to minimum performance bounds offered by fog providers, who need to monitor the dynamic resources, such as throughput and response time. The fog providers then check whether violation occurs, e.g., response time is higher than the agreed level. If such violations affect the profits of fog users, the fog provider has to compensate the users' losses. Several studies tackle the SLA-driven resource allocation problems on the simpler cloud platforms. Guaranteeing service levels is inherently difficult for fog providers, because of the dynamicity and heterogeneity of the minions. The same concerns also complicate the objective of satisfying QoS requirements.

Security/Privacy. Leveraging a multimedia fog platform requires the fog users to move their data to minions owned by the fog providers, third-party companies, or general publics. Exposing the data to others leads to security and privacy issues. Some of these have been studied for cloud computing users, including intrusion, data integrity, auditing, control, and availability. These issues also concern fog users. In addition, multimedia fog platforms also need to protect the security and privacy of end device owners. Hence, these are serious concerns in multimedia fog platforms, and need to be carefully considered.

Correctness/Fraud. In multimedia fog platforms, fog providers have to check the correctness of outputs from minions. Moreover, some malicious minions may send fake outputs to cheat for higher rewards. Because the resources in cloud platforms are fully controlled by cloud providers, there is no fraud issue in the cloud platforms. The problem of fraud has been studied in the crowdsourcing community [Vuurens and Vries 2012, Li et al. 2014], e.g., relevance judgment approaches are proposed to compare the outputs from different workers to detect cheaters. The multimedia outputs of our fog platform are huge (video files, for example) and diverse (different media types, for example), and thus it is harder to determine the correctness and detect the fraud. Avoiding fraud and ensuring correctness must be done for the sustainability of multimedia fog platforms.

Last, we emphasize that the above-mentioned challenges are representative ones, but by no means exhaustive. Indeed, there are more open challenges in emerging multimedia fog platforms, which lead to many research opportunities.

10.4 Distributed Multimedia Applications: What We Can Learn from Prior Studies

We argue that executing distributed multimedia applications in the fog is not a totally new idea: *multimedia researchers have been leveraging resources from machines in the cloud and crowd for optimizing massively distributed multimedia applications in the past*. In the following, we summarize some of our prior and ongoing projects that capitalize resources of machines across data centers, edge networks, and end devices. Readers may find new insights in these studies for building a general multimedia fog platform. Furthermore, we also point out open issues that may result in potential future work.

10.4.1 CrowdMAC: A Network Sharing Framework

More and more people use mobile Internet in their daily life. Therefore, the dataplan service providers deploy costly infrastructure to achieve the required QoS. However, the dataplan contracts often last for years, and thus are inflexible. This results in low dataplan quota utilizations for some users, and may drive other potential users away. Ericsson reports that the average unused, *wasted* monthly traffic quota is up to 61% (<http://tinyurl.com/zpegzdr>), causing frustration for mobile users (<http://tinyurl.com/z7kry7j>). The same report also indicates that up to 32% of mobile users exceed their monthly traffic quotas, and are charged at much higher rates. Three observations can be made from the above statistics: (i) light mobile users may find the fixed-term contracts not appealing, (ii) heavy mobile users may accidentally exceed the traffic quotas and be charged at higher rates, and (iii) other mobile users could end up with residual monthly traffic quotas.

In Do et al. [2012, 2016], we propose an on-demand mobile Internet system utilizing the residue dataplan resources of mobile Internet users (mobile hotspots) to help other mobile users (mobile clients) who need on-demand network connections. Figure 10.5 illustrates a possible usage scenario of CrowdMAC. In particular, mobile clients hire nearby mobile hotspots to send/retrieve data to/from the Internet. Mobile clients and hotspots communicate with each other using local (one-hop) wireless networks, such as WiFi Direct or Bluetooth. Using such a trading mechanism has many advantages; for example: (i) it provides connectivity to unconnected mobile clients and (ii) it shortens the delay of data transfer through the cellular connections of nearby devices. Mobile hotspots can, therefore, potentially be compensated for unused monthly traffic quotas.

At first glance, the concept of sharing mobile network access seems to be straightforward. However, designing such a system is not an easy task because



Figure 10.5 A usage scenario of CrowdMAC for shared mobile network access.

of the following complex trade offs: (i) accepting more requests leads to higher revenues, but may result in buffer overflow and long delay, and (ii) different mobile Internet users provide different QoS guarantees at diverse prices. CrowdMAC can be seen as a potential application of our multimedia fog platform using freelance minions (mobile hotspots). Building such a marketplace-inspired crowdsourcing framework involves several of the fog challenges shown in Figure 10.4. First, in terms of its business model, the system should support incentives for mobile hotspots to share their connectivities. Sharing connectivities will increase the delay and energy consumption of mobile hotspots. Therefore, designing a good business model is challenging and important to attract mobile hotspots. Second, in terms of SLA/QoS, a mobile hotspot must be capable of serving multiple clients while ensuring good connection quality for the concurrent users. This is essential to attract mobile clients to participate in the transfer. Third, in terms of resource management, schemes that enable mobile clients to leverage multiple mobile hotspots for faster data transfers is inherently hard due to the system dynamics.

Other important features of CrowdMAC include error recovery from connection losses among mobile devices, energy saving at mobile clients and hotspots, and secure data transfers. We focus on admission control and mobile hotspot selection problems Do et al. [2012, 2016]. A mobile hotspot upon receiving a request from a mobile client for data transfer needs to decide if it should admit and serve the request, based on factors such as whether it currently incurs any background traffic or if it may be busy with serving requests from other clients. We design an admission

control algorithm for mobile hotspots, and the goal of the algorithm is that the mobile hotspot can maximize its earned credits, while providing a guaranteed data delivery delay for each request it admits. A mobile client with a large content requires high bandwidth for fast data transfer. In our scheme, we allow the client to aggregate bandwidth from connections of multiple nearby hotspots, if available. More specifically, the client divides the content into multiple segments, and for each segment it hires a nearby mobile hotspot for transport. Therefore, the whole content can be transferred through multiple connections at the same time (known as multihomed). We solve the problem of selecting mobile hotspots for individual segments to transfer the whole content efficiently using the Lyapunov optimization framework [Georgiadis et al. 2006]. The details of the algorithms are given in our earlier work Do et al. [2012, 2016].

We develop a prototype system and evaluate it using upload scenarios. Mobile devices form a local P2P wireless network. There are four Android phones and two laptops (not shown in the figure) used in our testbed. Depending on the experiments, we configure them in one of the two scenarios: (i) one hotspot and several mobile clients as shown in Figure 10.6(a) and (ii) one mobile client and several hotspots as shown in Figure 10.6(b). The two laptops are used to generate WiFi background traffic. We examine our prototype at different locations: our lab, a cafe, and a park, where different cellular data rates are observed. We report sample results from the lab in the following.

Mobile hotspots earn credits from mobile clients by transferring data for clients. In general, each mobile hotspot determines its own function for charging clients. In the evaluations, we choose a common method: mobile hotspots earn credits

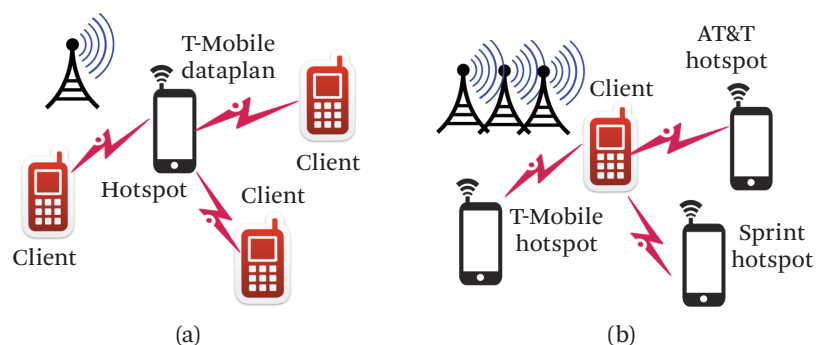


Figure 10.6 Our Android-based testbed: (a) with one hotspot and three mobile clients and (b) with one mobile client and three hotspots. (From Do et al. [2016])

proportionally to data size to compensate their dataplan, energy consumption, and SLA fee. We consider the following performance metrics:

- *Request delay*: Delay to complete a segment transfer, i.e., the time difference between a mobile hotspot admitting a request and the last packet of that segment being transmitted to the Internet.
- *Content delay*: Delay to complete the transfer of the content (which is divided into multiple segments), i.e., the time difference between the first request being issued by a mobile client and the last packet of the content being transmitted to the Internet.
- *Revenue*: The total credits a hotspot earns.
- *Cost*: The credits a client pays for content transfer.
- *Energy usage*: The consumed energy amount.

Evaluations of the Delay Bounded Admission Control Algorithm

We select one device to be the mobile hotspot, and vary the number of clients in this experiment (see Figure 10.6(a)). Figure 10.7(a) presents the average request delay when there is only one client. With the same δ_h (incoming rate of requests), increasing V_h (importance of revenue for a hotspot h , a control parameter) leads to higher average request delay. For example, with $\delta_h = 10$, the average request delay increases from 8.8 s at $V_h = 100$ to 41.8 s at $V_h = 5000$. This is because the mobile hotspot admits and simultaneously serves more requests with larger V_h . Given the same V_h value, a larger δ_h provides lower average request delay. The reason is that larger δ_h leads to fewer admitted requests, shorter service rounds, and lower average request delay.

We now investigate average request delay in a scenario with three clients. The results are depicted in Figure 10.7(b). The trend is consistent with the observations mentioned above. We also calculate the maximum request delay and maximum service round at $\delta_h = 10$, and report them in Figure 10.7(c) for different values of V_h . We observe that the maximum request delay is always smaller than the maximum service round, and the maximum service round is bounded by the worst case delay bound. More specifically, the ratio between the maximal request delay and the delay bound is about 80%; the ratio between the maximal service round and the delay bound is about 95%. This validates the correctness of our algorithm.

Evaluations of the Mobile Hotspot Selection algorithm

In this experiment, one device works as a mobile client, and uploads content via three hotspots with diverse prices. Mobile hotspots are set with large $V_h = 2000$ and

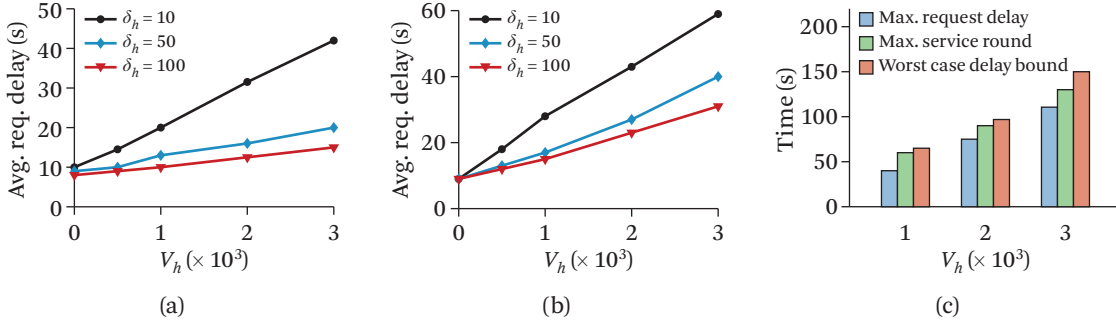


Figure 10.7 Performance of the Admission Control algorithm: (a) average request delay with one mobile client, (b) average request delay with three mobile clients, and (c) maximum request delay, service round, and delay bound with three mobile clients and $\delta_h = 10$. (From Do et al. [2016])

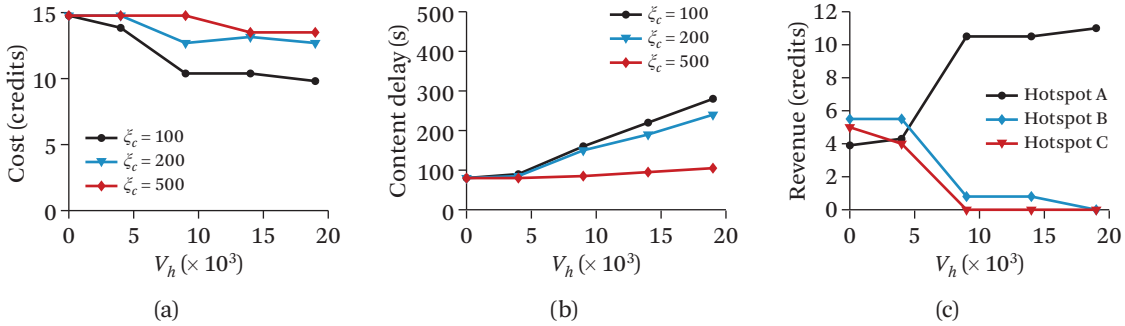


Figure 10.8 Performance of the Hotspot Selection algorithm: (a) cost paid by the client, (b) content delay, and (c) revenue of each mobile hotspot. (From Do et al. [2016])

small $\delta_h = 10$. Figure 10.8 presents our results. Given the same ξ_c (sending data in each time slot of client c), with higher V_c (cost weight for c), the client pays lower cost (as shown in Figure 10.8(a)), but experiences longer content delay (as shown in Figure 10.8(b)). Figure 10.8(c) depicts that at $V_c = 20000$ and $\xi_c = 100$, the client always picks Hotspot A, which is the hotspot with the lowest cost. This selection causes high content delay, which increases from 62.6 s at $V_c = 100$ to 271.8 s at $V_c = 20000$. These observations reveal a trade-off between cost and content delay at the mobile client. With the same V_c , larger ξ_c values lead to smaller content delays. For example, at $V_c = 20000$, content delay is 103.8 s with $\xi_c = 500$, but is 271.8 s with $\xi_c = 100$.

10.4.2 Smartphone-Augmented Infrastructure Sensing

Infrastructure sensing systems with in-situ sensors improve *situation awareness* by allowing individuals to *query* the environments for events of interest. For example, before going out in Paris, a tourist might want to know whether the Eiffel Tower is crowded or not. By analyzing a few frames of the surveillance video captured at the Eiffel Tower, we can estimate how many people are there, and determine whether it is crowded. Although infrastructure sensing is popular, deploying, managing, and maintaining all the in-situ sensors is a very expensive, labor-intensive, and error-prone process. Therefore, infrastructure sensing systems often suffer from *inaccurate* or/and *incomplete* sensory data due to the limited resources. For example, a community that can only afford a few cameras has to install them at major intersections, leaving many spots uncovered, which compromises public safety.

In Liao et al. [2014], we present a system, called Smartphone-Augmented Infrastructure Sensing (SAIS), using sensor-rich smartphones to finish tasks. The state-of-the-art smartphones come with many sensors, such as GPS readers, accelerometers, gyroscopes, microphones, cameras, and network interfaces. These smartphones allow us to augment infrastructure sensing by incorporating *crowd-sensing* for collecting more sensory data at lower costs. Crowdsensing [Ganti et al. 2011] also refers to collecting this sensory data from many smartphones. Through information gathered by civilians and officials, we build a mobile dashboard that provides a sense of situation, e.g., answering questions from a tourist as shown in Figure 10.9. The reported situation may influence potential actions that a user or an operator may take or adapt in order to avoid dangerous events, such as fights, riots, protests, demonstrations, fires, chemical leaks, stampedes, and high crowd levels. In other words, all smartphone users use the SAIS dashboard to produce and consume information about safety events, and help one another to create better situation awareness.

SAIS can be seen as a sample application of our multimedia fog platform leveraging the sensors installed on freelance minions (smartphones). SAIS faces some fog-related challenges shown in Figure 10.4. First, in terms of resource management, because of the limited resources and high mobility of smartphones, managing the resources to minimize resource consumption while achieving acceptable sensing accuracy is difficult. Second, in terms of correctness/fraud, the problem is further complicated by the fact that a single smartphone user may fail to satisfy the accuracy requirement, and multiple smartphone users may be needed. Determining the minimum number of workers to assign to each job in order to meet the accuracy requirement is hard. Third, in terms of business model, because we try to utilize the smartphones owned by the crowds to help us collect sensory data, it is not easy to convince them to contribute/sell their precious resources.

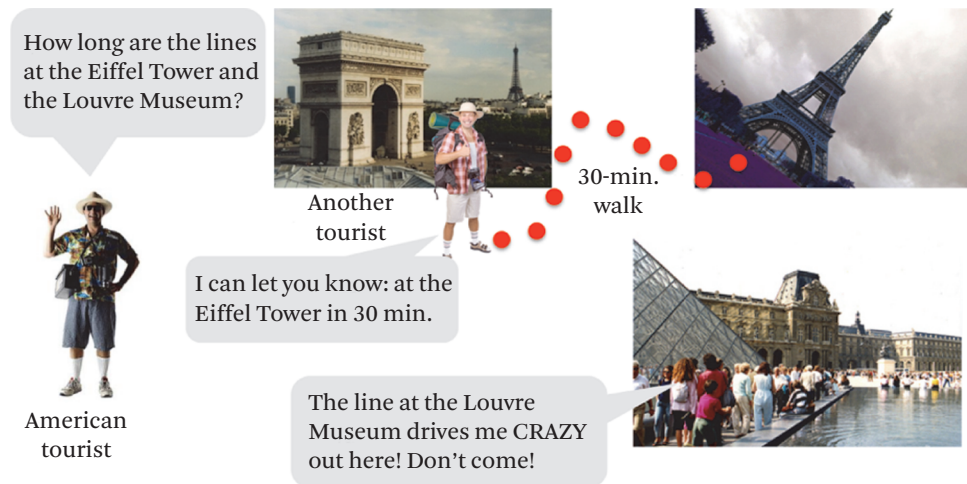


Figure 10.9 A usage scenario of smartphone-augmented infrastructure sensing.

We focus on the first challenge and propose to employ a centralized server for: (i) governments, companies, or individuals to submit on-demand sensing requests, and (ii) smartphone users to accept spatial-temporal specific sensing jobs [Liao et al. 2014]. The crux of SAIS is a job assignment algorithm, which minimizes the energy consumption while maintaining high sensing accuracy. We propose an optimal algorithm and an efficient heuristic algorithm to solve the job assignment problem. The optimal algorithm gives the optimal task assignment, but may lead to long running time even for small-scale problems. Therefore, it seems impractical to employ the optimal algorithm in a real system, in which the algorithm has to serve tens of thousands of queries and users. We therefore develop an Efficient Task Assignment (ETA) algorithm in the following.

The design principle of ETA is as follows. We define $\lambda_{w,l}$ as the utility of the worker w performing tasks at location l , which is a ratio of the number of covered sensing tasks to the energy consumption. We iteratively assign tasks, which consist of the target location and the sensor to turn on, to a worker. We assign the tasks at location l to worker w who has the largest utility $\lambda_{w,l}$, as long as worker w will not exceed his/her energy budget. ETA checks whether the required error bound of sensors is satisfied in each run, and the satisfied (completed) queries are deleted from the query set. The iteration is terminated if all the queries are satisfied or the workers cannot perform any more tasks. In SAIS, if the moving cost has a higher weight than the sensing cost, the algorithm tends to assign tasks that are closer to workers; and vice versa. The details of the problem formulation and the algorithms are shown in our earlier work [Liao et al. 2014].

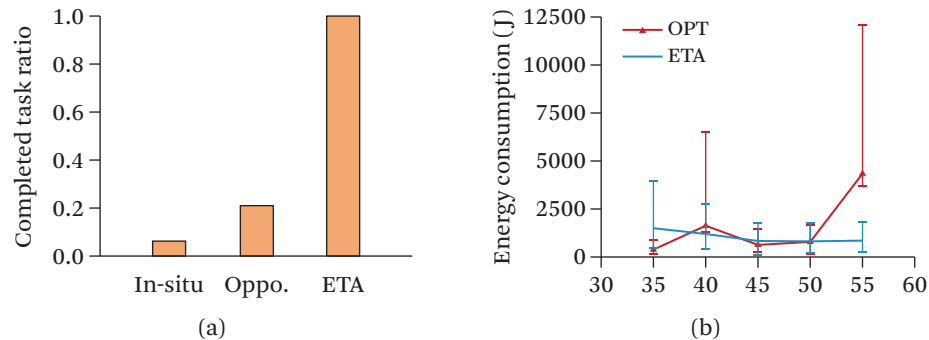


Figure 10.10 Advantages of ETA: (i) higher completed task ratio and (ii) more practical energy consumption. (From Liao et al. [2014])

We implement our task assignment algorithm and three baseline algorithms: (i) *in-situ sensors only*, (ii) *in-situ sensors with opportunistic sensing*, and (iii) *optimal*, in a simulator developed in Java. We adopt the random waypoint model to generate user movement patterns in our simulations. In order to understand the performance of our algorithms, we use: (i) completed task ratio, (ii) energy consumption, (iii) running time, and (iv) response time as our metrics. The response time is defined as the time difference between receiving a query and when the query is satisfied.

Advantage of combining in-situ sensing and crowdsensing. Figure 10.10(a) shows that in-situ sensors can only cover 6% of queries, and smartphone users can significantly improve the coverage of the system. The in-situ sensors with opportunistic sensing achieve a coverage of 20%, and ETA instructs smartphone users to move to the required locations and thus covers all queries. We no longer consider in-situ sensing in the following simulations.

The optimal algorithm is less practical in large-scale systems. The optimal (OPT) algorithm is implemented in CPLEX (<http://tinyurl.com/j78vao4>) with a 1-min time limit, due to the realtime nature of the considered problem. Figure 10.10(b) reports the energy consumption of OPT and ETA with 40 queries. This figure shows that ETA outperforms OPT when the number of workers exceeds 50. Hence, ETA is more practical, while the OPT algorithm may be used in smaller systems.

ETA outperforms the in-situ sensors with opportunistic sensing in terms of completed task ratio. Figures 10.11(a) and 10.11(b) present the results of the completed task

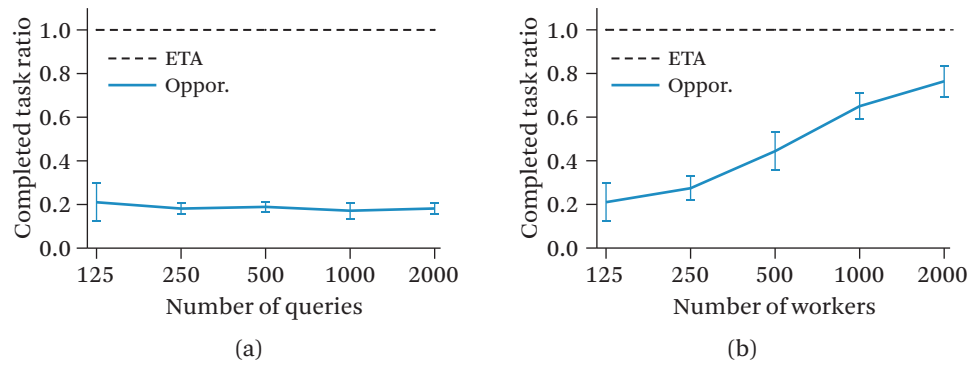


Figure 10.11 Completed task ratio of ETA and opportunistic sensing: (i) diverse numbers of queries and (ii) diverse numbers of workers. (From Liao et al. [2014])

ratio of ETA and in-situ sensors with opportunistic sensing. The results show that ETA achieves a high completed task ratio (100%) with different numbers of queries and workers. However, in-situ sensors with only opportunistic sensing cover less than 20% and the ratio gradually decreases when the number of queries increases. The low completed task ratio significantly harms the user experience since most of the queries are not satisfied. To avoid degraded user experience, the in-situ sensors with opportunistic sensing system must recruit more workers. However, the number of required workers may be staggering; for example, as shown in Figure 10.11(b), the in-situ sensors with opportunistic sensing system requires 1000 workers to complete 100% queries, while ETA only needs 125 workers (12.5% of the work force size).

ETA outperforms opportunistic sensing in terms of response time. The response times of the two algorithms are given in Figures 10.12(a) and 10.12(b). Since ETA can instruct workers to get to the required locations, the response time of ETA is always less than 1000 seconds. However, opportunistic sensing cannot instruct workers to perform sensing tasks, and suffers from 6 times higher response time. If the number of queries increases, the probability of smartphone users who occasionally cover the required locations also increases. Therefore, the response time of opportunistic sensing is gradually decreasing while the number of queries is increasing, as illustrated in Figure 10.12(a). Increasing the number of workers, however, does not reduce the response time for opportunistic sensing. This is because the density of workers is still too low, and it takes the workers some time to run into the locations of some queries.

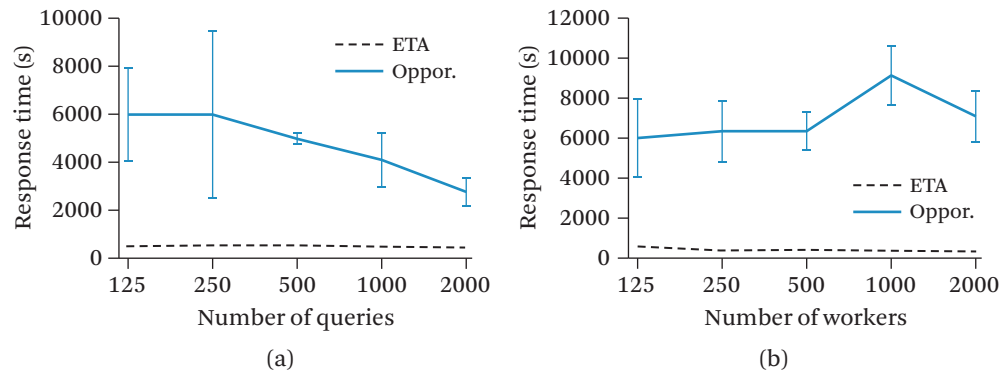


Figure 10.12 Response time with diverse number of: (i) queries and (ii) workers (From Liao et al. [2014])

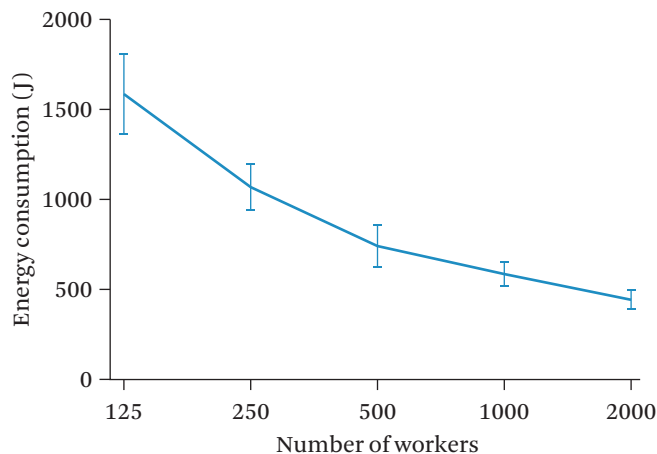


Figure 10.13 Energy consumption of ETA with diverse numbers of workers. (From Liao et al. [2014])

More workers result in lower total energy consumption. Next, we vary the number of workers, and report the total energy consumption of ETA in Figure 10.13. The energy consumption includes both sensing energy and moving energy. This figure shows that more workers lead to lower energy consumption. A closer look indicates that such reduction comes from the fact that more workers allow ETA to send closer workers to individual query locations, saving a large amount of moving energy. We also find that ETA has a very short running time: only 4 seconds with 200 workers. This renders it suitable for practical systems.

10.4.3 A Crowdsourced Animation Rendering Service

Animation studios generate huge and fluctuating computational demands for rendering high-resolution pictures and high-quality video sequences, and thus hosting dedicated servers is cost-ineffective. From 2011 to 2014, the requirement to render a modern animation increased four times. More specifically, 12500-core and 55000-core rendering farms were required by *Cars 2* and *Big Hero 6* in 2010 and 2014, respectively (<http://tinyurl.com/zl6etd6>). Such a high-end rendering farm may not be affordable for small animation studios and start-up companies. Hence, these companies adopt on-demand cloud rendering platforms, such as Shaderlight (<http://limitlesscomputing.com/Shaderlight>), RenderStorm (<http://www.render-storm.com/>), and ZYNC (<http://www.zyncrender.com/>) to reduce the costs. However, the cost is still too high to the small companies, e.g., rendering a 10-second 1080P animation with 300 frames costs 700 dollars using Shaderlight. Moreover, a shot may be rendered multiple times before being finalized, so that rendering an animation with cloud rendering platforms is still too costly.

With the distributed solution offered by a collaborated fog rendering company, animation studios have access to less expensive on-demand resources. The on-demand resources are rented from idling resources of desktops and laptops owned by the crowds. Realizing such a service for maximum profits would require several immediate problems to be solved, such as: (i) dividing each rendering request into manageable rendering jobs, (ii) estimating the completion time of each rendering job, and (iii) determining the reduced carbon footprint due to the multimedia fog platform. Figure 10.14 illustrates the fog rendering platform, which consists of client applications on end devices for rendering, Web interface for animation studios to submit their requests, and a server to monitor and aggregate the rendered results.

Implementing and optimizing the platform requires us to consider *all* the challenges shown in Figure 10.4. First, in terms of business model, we need to design it to (i) motivate crowds for contributing their idling resources, (ii) earn more profits, and (iii) reduce the costs on animation studios. Second, in terms of resource management, we need to predict available resources and manage the resources of the laptops/desktops owned by dynamic crowds. Third, in terms of SLA/QoS, we need to predict the completion time of each rendering job to guarantee the SLA, i.e., the animation studios will ask for the precise estimated completion time. Fourth, in terms of security/privacy, we need to convince the crowds that rendering animations on their desktops will not invade their privacy. Moreover, we need to convince animation studios that the animations rendered on crowds' desktops will not be leaked. Fifth, in terms of virtualization, different animation rendering studios use different rendering engines. It is not possible to install all of them on the crowds'

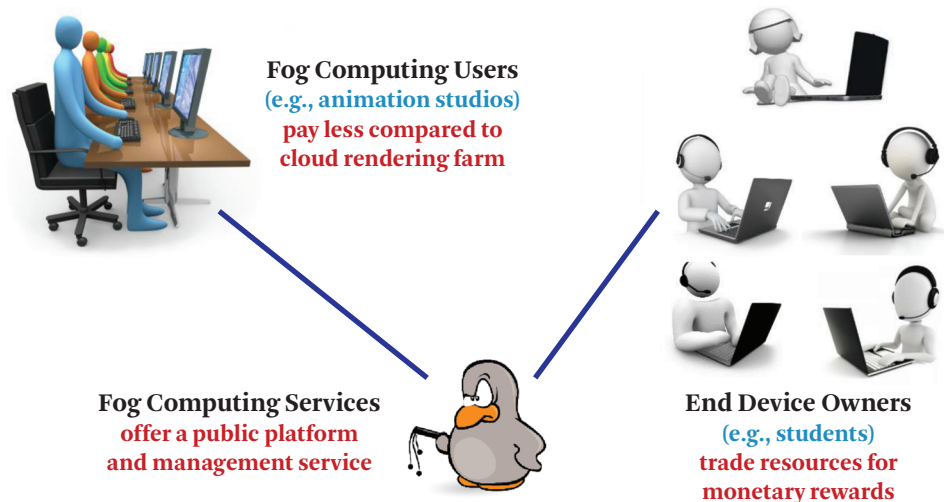


Figure 10.14 A multimedia fog platform based rendering farm.

machines. Hence, using virtualization technologies for dynamically changing the rendering engines is important. Moreover, virtualization is also one solution to cope with the privacy issue. However, running virtual machines on desktops incurs high overhead. Hence, a lightweight virtualization technology is required. Last, in terms of correctness/fraud, we have to prevent malicious crowds from submitting wrong results.

Taking SLA as an example, we analyze 32,704 real rendering jobs collected by the company, and compute the *normalized deviation* as the prediction error normalized to the actual job completion time. Currently, the company employs linear regression to predict the job completion time. We give the cumulative distribution function (CDF) of the normalized deviation in Figure 10.15(a). This figure reveals that more than 80% of the predictions with linear regression deviate from the actual job completion time by more than 100%. Hence, a better estimation algorithm is required in this application. The completion prediction problem is a regression problem, which can be solved by many existing machine learning approaches. We proposed an algorithm inspired by the existing machine learning approaches. Figure 10.15(b) shows sample prediction results from the proposed algorithm, which reveals that only 20% of them deviate from the actual job completion time by more than 100%. This is a large improvement over the current practice used in commercial fog computing rendering farms. More details on our prediction algorithms are presented in our prior work [Hong et al. 2016a].

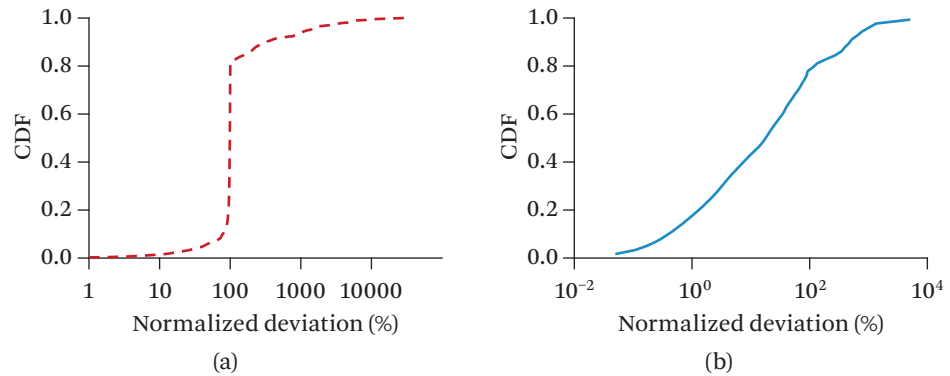


Figure 10.15 The normalized deviation of: (a) the linear regression based algorithm (From [Hong et al. \[2016a\]](#)) and (b) our proposed algorithm.

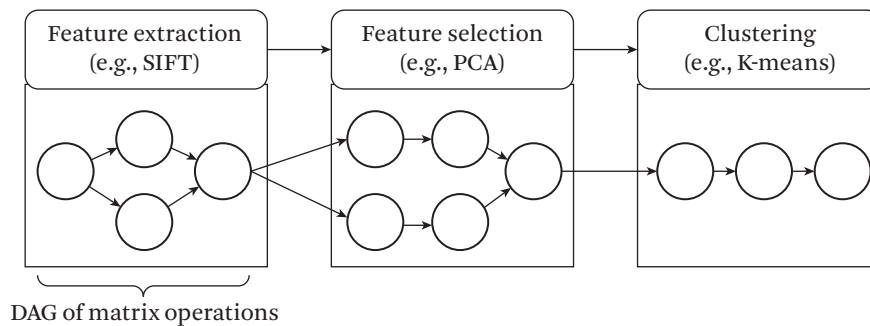


Figure 10.16 Typical steps of image clustering applications.

10.4.4 Scalable and Distributed Principal Component Analysis

Much multimedia data is acquired sequentially over time, including video feeds from surveillance cameras and time series from wearable cameras and sensors. In these settings, rather than waiting for all the data to be acquired before performing our analysis, data streams are continuously processed as soon as they become available.

Analyzing video data is often done by applying machine learning algorithms. Figure 10.16 shows the steps for an image clustering application where an image is automatically divided into a set of categories (e.g., people, vehicles, or landscapes). The application starts with feature extraction. A common example of image feature extraction is Scale Invariant Feature Transform (SIFT) [[Lowe 2004](#)], which

represents the interest points in a video frame that are invariant to image transformations, such as scaling and rotation. The extracted features are reduced to pick the most discriminative ones that have the highest variance among the data. This is typically done using principal component analysis (PCA). Finally, the images are clustered into a set of categories according to the reduced set of features. Each of the algorithm steps in Figure 10.16 is typically composed of a set of matrix operations represented as a direct acyclic graph (DAG). Each DAG node represents a process that performs a matrix operation on the input stream and each edge represents data flow from one operation to another. In fog environments, some matrix operations on the DAG can be computed on end devices and a subsequent part can be computed in data centers. The decision is based on the computation time on the minions which typically have bounded resources. Since each matrix operation must wait for the entire intermediate data produced by its predecessor, a large amount of intermediate data will introduce delays and limits the opportunity to leverage the computational resources in the cloud. In the following, we show a method to reduce intermediate data in the context of an algorithm called sPCA [Elgamal et al. 2015], which performs PCA (middle step in Figure 10.16) in an optimized manner to reduce the intermediate data that flows from one operation to another. Such optimization tackles several fog-related challenges shown in Figure 10.4, including resource management, SLA, and QoS.

Background. Given a matrix Y of size $N \times D$ (N rows and D columns), a PCA algorithm obtains d dimensions ($d \leq D$) that represent the most variance (and hence information) of the data in matrix Y . The d dimensions are a linear combination of the original D dimensions and they are known as the principal components. For example, in the image processing domain, PCA is used to obtain the principal facial components whose linear combination could recreate any face in the image dataset. In information retrieval, the principal components represent the principal terms in a set of documents.

Execution Model. We note that most PCA algorithms work in multiple synchronous phases, and the intermediate data is exchanged at the end of each phase. Some phases can be computed on the weak minions on end devices, and subsequent phases can be computed on powerful minions in data centers. For the simplest case, we assume that we have multiple end devices and a data center. Since each phase must wait for the entire intermediate data produced by its predecessor phase to be received before its execution starts, large intermediate data introduces delays and can become a major bottleneck even if both phases run on minions in the same

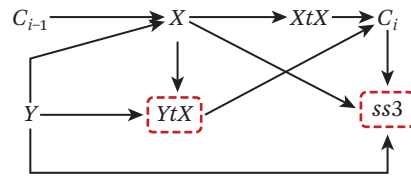


Figure 10.17 The job graph of sPCA. Nodes are labeled with variable names. Dotted rectangles indicate distributed jobs. The input to the algorithm is matrix Y and the output is C , which contains the principal components. C_{i-1} is the principal components obtained from the previous iteration.

data center. The delay gets even larger when intermediate data is exchanged among minions on end devices and in data centers that are geographically separated. The exact delay will depend on network topology, link speed, and I/O speed, as well as the software platform used to manage the data center and run the PCA code. Some software platforms, e.g., Hadoop/MapReduce, exchange intermediate data through the distributed storage system, while others, e.g., Spark, exchange data through shared virtual memory. For the analysis of intermediate data to be general, we consider the total number of bytes that need to be exchanged, and we abstract away the details of the underlying hardware/software architecture.

sPCA. sPCA is a scalable algorithm for computing PCA for large datasets on the fog platforms. We focus on one major contribution of our work [Elgamal et al. 2015], which is reducing the intermediate data that flow from one operation to another due to the space limitations. Figure 10.17 depicts a part of the DAG that shows dependency between matrix operations in sPCA. Each node is labeled with the variable that the operation produces. A link from node Y to node X indicates that data of variable Y must be computed before starting the job that computes variable X . Variables carried over from the previous iteration are distinguished with the index i . Variable Y is the input to the algorithm. The output (the principal components of Y) is C_i . For the first iteration ($i = 1$), C_0 is initialized randomly from a normal distribution. Then, matrix C_{i-1} and the input matrix Y are used to generate the intermediate matrix X . Matrix X is used for three other computations. First, it is used to create matrix XtX , which is the product of the transpose of X with X . Second, X is used to produce YtX , which is the product of X with the transpose of input matrix Y . Third, X is used to compute $ss3$, which is a scalar value that is used to compute the variance of matrix C_i .

sPCA employs a heuristic to decide which jobs are executed on which minions based on different amounts of resource requirements. If all the inputs of one

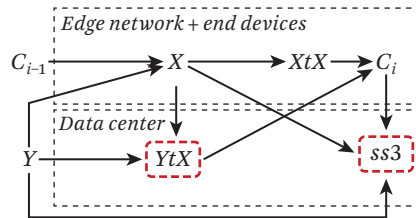


Figure 10.18 Division of jobs between the end devices, edge networks, and data centers.

operation can fit into the memory of the minions in edge networks and on end devices, the computation is done on these less-powerful minions; otherwise the data is transferred to the data center to do the computations in a distributed manner. Taking the *Tweets* [Elgamal et al. 2015] dataset as an example, the dataset consists of more than 1.2 billion rows representing the tweets and 71,503 columns representing the distinct words. The sparse representation of this dataset occupies 94 GB. With each end device equipped with 128 GB memory, we initially found that only two jobs can be computed, because they operate on two large matrices that cannot fit in the memory of a single machine. Figure 10.18 shows the division of jobs among the data center, edge networks, and end devices. The upper dashed rectangles enclose jobs that are computed on the minions in the edge networks and on the end devices, and the lower dashed rectangle encloses jobs that are computed in the data center. Arrows that pass from the upper to the lower rectangle indicate data transfer from the edge networks and end devices to the data center.

Through analysis of the DAG, we figure out that the intermediate matrix X is a large intermediate matrix that has to be fed to three different operations, two of which are large-scale operations that require executions in the data center. Although input matrix Y is a sparse matrix for several large-scale datasets, X is typically a large dense matrix which can become a major scalability bottleneck. sPCA solves that issue through *redundant computation*. First, we note that while storing and exchanging X is expensive due to its large size, computing it is a relatively lightweight operation when we use sparse matrix multiplication. To leverage this property, we redesign the algorithm by redundantly recomputing X at each job that consumes it as input. Figure 10.19 illustrates this optimization. The figure shows that although X is computed multiple times, the data passing among the distributed minions is only limited to C_i , which is a small matrix (30 MB for the *Tweets* dataset).

Evaluations. The implementation of sPCA has one main driver program that runs on a single machine that needs not be too powerful. The driver program issues dis-

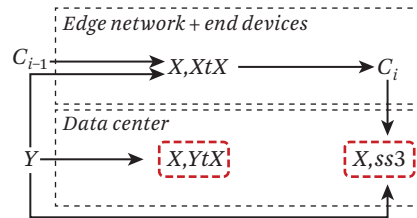


Figure 10.19 Modified job graph that shows our optimization of reducing intermediate data by redundantly computing X in the three jobs.

tributed operations in a data center with 8 Amazon EC2 m3.2xlarge instances. We implement the distributed operations on MapReduce [Dean and Ghemawat 2008]. We use a large dataset for the comparisons. We construct a matrix such that the rows represent the tweets and the columns represent all words that appear in each tweet. The matrix is of size $1,264,812,931 \times 71,503$, which occupies about 94 GB. In the following, we first evaluate the improvement of reducing the intermediate data that flows from the end device to the data center, then we compare sPCA against the closest counterpart on MapReduce that is implemented in the Mahout library (<http://mahout.apache.org>); we refer to such an implementation as *Mahout-PCA*.

We use a subset of the *Tweets* to evaluate the improvement of reducing the intermediate data that flows among minions. The dataset consists of 100,000 rows and 71,503 columns. We measure the running time of sPCA with and without reducing intermediate data X . The results show that it takes 7 seconds to compute X and YtX from the input matrix Y , while it takes 54 minutes to compute YtX from the intermediate matrix X . This shows that the overhead of loading X on minions is significantly slower than recomputing it. Therefore, recomputing it on minions significantly speeds it up compared to transmitting it across minions.

We compare the sPCA against the Mahout-PCA with respect to the intermediate data size, which is the amount of data generated by each algorithm during its execution. We note that in many cases the intermediate data generated by the algorithm far exceeds the size of the input data, and thus becomes a major bottleneck. Our results show that sPCA generates much smaller intermediate data compared to Mahout-PCA. Figure 10.20 shows that Mahout-PCA generates 961 GB of intermediate data, whereas sPCA-MapReduce produces 131 MB of such data, a factor of $3,511\times$ reduction.

Next, we compare the running times of sPCA and Mahout-PCA. A sample of the results is shown in Figure 10.21 for the *Tweets* dataset. Other results are similar [Elgamal et al. 2015]. In this figure, we vary the number of rows in the input matrix, but we use the same number of columns, namely the full 71,503 columns of the

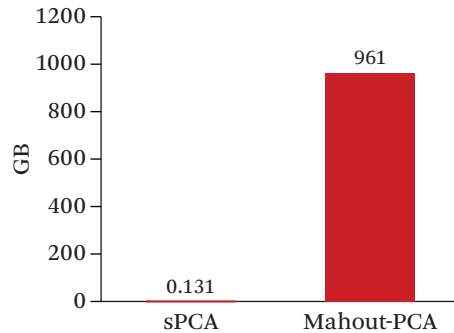


Figure 10.20 Size of the intermediate data of sPCA vs. Mahout-PCA. (From Elgamal et al. [2015])

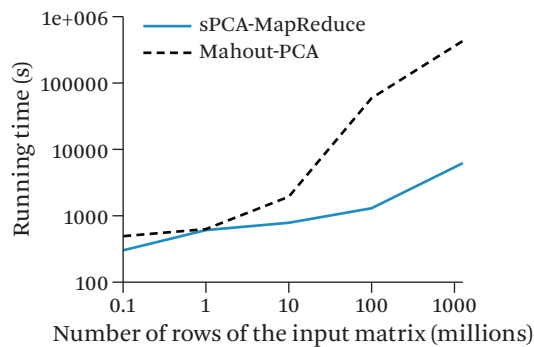


Figure 10.21 Running time with diverse numbers of rows. The axes are in log scale. (From Elgamal et al. [2015])

dataset. The results in Figure 10.21 show that the running times for both algorithms are close for small datasets (i.e., up to 10 million rows). For larger datasets, however, sPCA-MapReduce is two orders of magnitude faster than Mahout-PCA. The reason for this is that the benefits of the optimizations in sPCA pay off better when we scale to larger datasets.

10.5 Deployment: Open-Source Platforms

The above projects are all potential applications of multimedia fog platforms. However, we often re-implement the applications from scratch, which is tedious, error-prone, and inefficient. Therefore, we propose to build and optimize a general fog platform for distributed multimedia applications. With such a platform

ready, we can avoid reinventing wheels in future projects. More specifically, we believe that it is important to design, implement, and deploy a general fog platform for: (i) communication-, (ii) computation-, (iii) storage-, and (iv) sensor-bound distributed multimedia applications. Our considered fog platform aims to capitalize all available resources in data centers, edge networks, and end devices. We believe that such a goal has always been part of multimedia system research, because multimedia applications are resource-hungry and delay-sensitive. The multimedia research community will be more involved in the development of a general, optimized, multimedia fog platform for all future distributed multimedia applications. In the following, we survey three existing open-source platforms, which can be extended into a multimedia fog platform.

OpenStack(<http://www.openstack.org>) is a famous and mature open-source management platform used in cloud computing. Cloud providers can use OpenStack to efficiently manage many computers and VMs in data centers. OpenStack provides several services for management, including Nova, Glance, Neutron, Swift/Cinder, Keystone, and Horizon. These services are responsible for managing VMs, images, networks, storage, security, and user interface, respectively. In addition to traditional VM technologies, such as Xen (<http://www.xenproject.org/>) and KVM (<http://www.linux-kvm.org/>), OpenStack also supports a lightweight virtualization technology, called LXC (<http://linuxcontainers.org>).

SaltStack(<http://saltstack.com>) is a lightweight management tool implemented in Python. SaltStack allows users to remotely configure a set of computers. SaltStack is developed by Google, and is integrated with Docker as its virtualization technology. SaltStack is also a possible choice to implement our multimedia fog platform, and is bundled with Kubernetes (introduced below).

Kubernetes(<http://kubernetes.io>) is developed by Google. It is a recent management tool designed for lightweight virtualization. Kubernetes has the same master-minion structure and it adopts Docker as its lightweight virtualization technology. The master of Kubernetes can manage all the minions and the Docker containers running on the minions. Kubernetes integrates SaltStack, but it is more comprehensive than SaltStack. More specifically, Kubernetes provides better fault tolerance, which automatically makes redundant containers and performs failure recovery—for example, if a user

requests to run Apache services. Kubernetes creates a pod consisting of multiple Apache containers. It then automatically selects appropriate minions to run the Apache containers. If one of the minions is broken or one of the containers is disconnected, Kubernetes automatically recovers from it.

Virtualization technologies such as Xen and KVM need entire guest operating systems, the necessary binaries, and libraries to create a virtual machine. It takes a powerful server to launch a virtual machine, which consumes a lot of storage space and computing power. Hence, these technologies are not suitable for our fog platform. Instead, containers such as LXC and Docker are used in several areas. Compared to virtual machines, containers require less resources and can be set up in a short time. More specifically, multiple containers running on the same host share the same operating system kernel, and use the namespaces to distinguish one from another. Modules running in containers are allowed just mandatory services they need, rather than including the full operating system. In summary, Docker and LXC containers are a lightweight virtualization technology, which is more suitable to be used on resource-limited minions, compared to traditional virtualization technologies.

Based on our survey on open-source platforms and virtualization technologies, Kubernetes is a state-of-the-art platform, consisting of SaltStack and the lightweight virtualization technology, Docker. We implement a Kubernetes-based fog platform, shown in Figure 10.22, which consists of three main components: (i) a user interface, (ii) Kubernetes, and (iii) Module Deployment Algorithm (MDA). Users send requests to the server through the user interface, and the server stores the requests in the request database. Each request may be split into several modules, which are packaged by Docker as container images. The container images are pushed to the devices and the corresponding modules are started from the container images. After collecting a bunch of requests, the server executes the MDA algorithm to make module deployment decisions, which are essentially the deployment plan of the corresponding modules of requests. The deployment plan is then sent to Kubernetes, which follows the plan and sends commands to deploy the modules on the devices. Finally, the results from the modules are stored in the result database and shown to the users through the user interface. More details about the MDA algorithm can be found in [Hong et al. \[2016b\]](#).

Figure 10.22 also reveals that multiple small modules can help one another to finish a request and send the results back to the server. In contrast, traditionally, we use a module to collect some sensing data and use another complex module

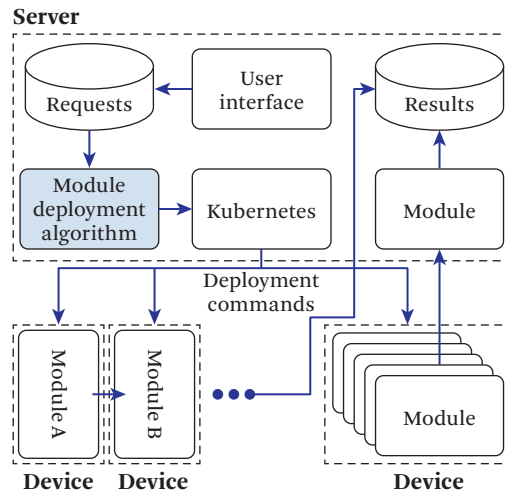


Figure 10.22 The Kubernetes-based fog platform. (From Hong et al. [2016b])

to process the sensing data on a powerful server. For example, if a user requests information on the crowdedness of a specific location, traditionally, we run a face detection module, that receives images from surveillance cameras and counts the number of humans on a server. In our proposed fog platform, we may split the same request into three smaller modules: (i) image collector, which is responsible for collecting images; (ii) feature extractor, which is responsible for extracting face features of the collected images; and (iii) face detector, which is responsible for counting the number of humans based on the extracted features. We then deploy these modules on multiple devices to share the load of a server and achieve lower latency.

We implement a testbed of the Kubernetes-based fog platform with Docker. In particular, we install Kubernetes on a mini PC with i5 CPU. We adopt Raspberry Pis (<http://www.raspberrypi.org>) as our minions and install both Kubernetes and Docker on Raspberry Pis. The Kubernetes installed on the mini PC is the master of all the Raspberry Pis. We use Wonder Shaper (<http://lartc.org/wondershaper/>) to throttle the bandwidth of the links between the devices and the links between the devices and the server. We assume that the devices communicate using WiFi mesh networks to stream data among the devices and connect to the Internet using 4G networks. The 4G network is only used for pushing container images to the devices, avoid high access fees. We limit the bandwidth between devices 300 Mbps, which is the upper bound of 802.11n. We also limit the bandwidth of 4G networks 150 Mbps.

We implement three modules: (i) image collector, (ii) face detector, and (iii) crowdedness monitor. The image collector module captures images using the camera sensor installed on Raspberry Pi. The face detector module counts the number of humans based on the image from the image collector module. The crowdedness monitor module measures the crowdedness based on the numbers from the face detector module. More specifically, there are three kinds of requests that can be by users, including (i) crowdedness, which uses all the three modules; (ii) number of humans, which uses the first and second modules; and (iii) image, which uses only the image collector module. For example, if a user requests the crowdedness in a cross street, we run the MDA algorithm and deploy these three modules on corresponding devices. The image collector module collects the image of the street and sends it to the face detector module. The face detector module receives the image, extracts the number of humans, and sends it to the crowdedness monitor. The crowdedness monitor module receives the detected number of humans and computes the crowdedness for the user.

The sizes of the container images of these modules are 121.9 MB, 251.6 MB, and 117.1 MB, respectively. The size of the container image of the face detector module is about 2 times that of others because the face detector module requires the OpenCV libraries, which need more space. However, the size all of these images is still smaller than a traditional VM image: a Windows 7 VM may require > 20 GB disk space. This is because the traditional VM contains the entire operating system, required libraries, and application binaries. The containers running on the same machine share the kernel with the host, and thus they consume less resources.

Figure 10.23(a) shows a photo of our testbed. We conduct an experiment to measure the overhead while we run multiple instances of the face detection application on the same host. Moreover, we compare the performance with/without containers. Figures 10.23(b)–10.23(d) report the consumed resources and processing time. The figures show that simultaneously running more than four instances of the face detection application on a Raspberry Pi leads to full CPU utilizations and increased processing time. This can be attributed to the fact that the Raspberry Pi has a quad-core ARM Cortex-A7 CPU, and our application is CPU-bound. Last, the figure also reveals that using containers results in virtually no overhead compared to directly running the applications on the host. More details on the testbed can be found in [Hong et al. \[2016b\]](#). Given the encouraging evaluation results, we are currently implementing several distributed multimedia applications on our multimedia fog platform.

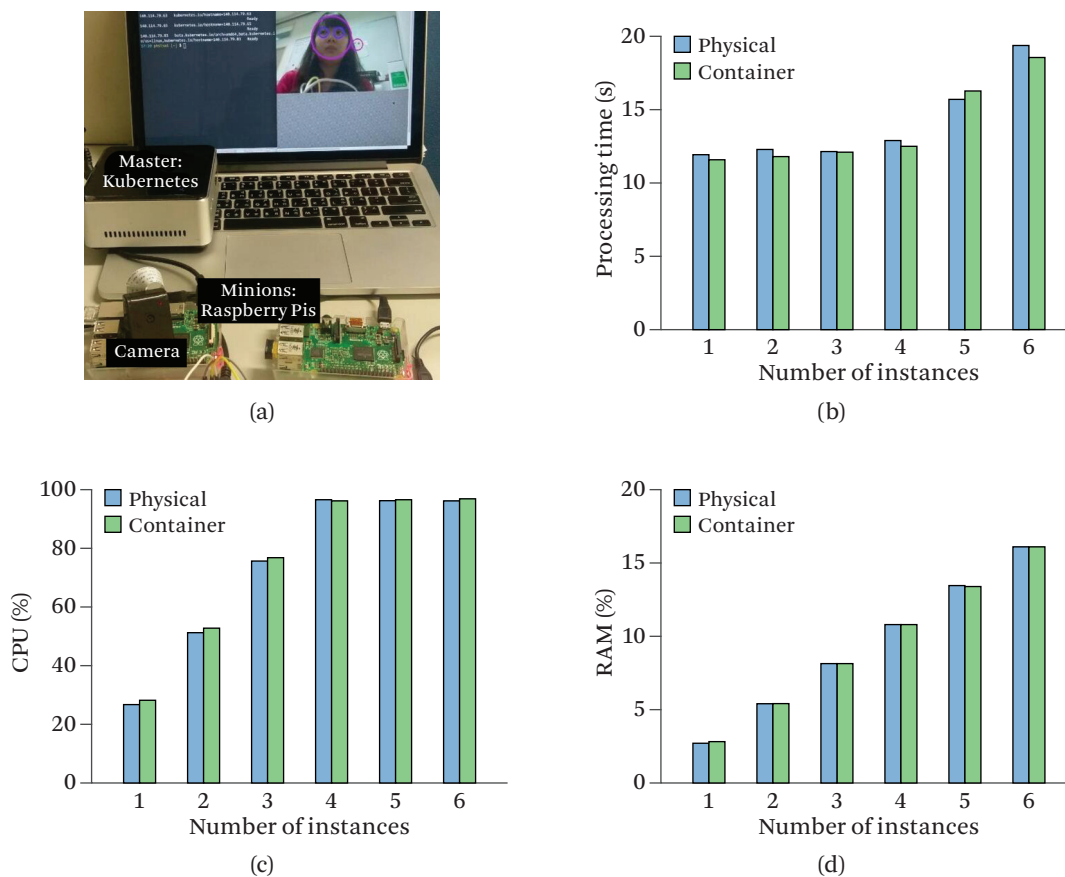


Figure 10.23 Implications of different numbers of application instances: (a) photo of the testbed, (b) processing time, (c) CPU utilization, and (d) RAM utilization.

10.6 Conclusion

In this chapter, we have proposed a multimedia fog platform, which provides communication, sensing, storage, and computation resources to build a low-latency and low-cost environment for distributed multimedia applications. We have discussed the following challenges of implementing the multimedia fog platform: (i) business model, (ii) virtualization, (iii) resource management, (iv) SLA/QoS, (v) security/privacy, and (vi) correctness/fraud. These challenges may stimulate future research in these exciting directions. Moreover, we have summarized four of our prior studies as examples of distributed multimedia applications. The

presented studies span over communication-, sensor-, resource-, and computation-bound multimedia applications. Each application considers all or some of the listed challenges (e.g., the animation rendering service on the multimedia fog platform considers all the listed challenges). Our prior studies solve some of the issues that occur when leveraging the resources from minions, but they are not running on a general multimedia fog platform. To avoid reinventing wheels, we have surveyed open-source platforms and have adopted Kubernetes and the Docker container to implement our general multimedia fog platform. Our preliminary experiment results show that using the Docker container does not lead to additional overhead compared with directly running the applications on minions. Hence, adopting Docker as the virtualization technology and using Kubernetes to manage them is promising for implementing the general multimedia fog platform.

This chapter is just a starting point for developing a general multimedia fog platform. There are a wide spectrum of open issues caused by the uncertainty and heterogeneity of minions; for example: (i) predicting the available resources, (ii) managing the heterogeneous resources, (iii) dividing the applications into smaller modules to fit the resource-limited minions, (iv) deploying the modules to the right minions for good QoS, (v) detecting the correctness of the results from the minions, and (vi) preventing fraud. We firmly believe the multimedia systems community has a unique opportunity to address these issues for a general, optimized multimedia fog platform.

Cloud Gaming

**Kuan-Ta Chen (Academia Sinica),
Wei Cai (The University of British Columbia),
Ryan Shea (Simon Fraser University),
Chun-Ying Huang (National Chiao Tung University),
Jiangchuan Liu (Simon Fraser University),
Victor C. M. Leung (The University of British Columbia),
Cheng-Hsin Hsu (National Tsing Hua University)**

Cloud gaming refers to a new way to deliver computer games to users via networks, where computationally complex games are executed on cloud servers, the rendered game scenes are streamed over the Internet to gamers with thin clients on heterogeneous devices, and the control events generated by input devices are sent back to cloud servers for interactions. In the late 2000s, we started to see cloud gaming services offered by startups, such as [OnLive](#), [GaiKai](#), and [Vbitus](#). Then GaiKai was acquired by Sony, which is a major game console developer [Sony-Gaikai](#). This was followed by competition between Sony's PlayStation Now and NVIDIA's Grid Game Streaming Service, which further heated up the cloud gaming market. In fact, a 2016 report from [Infiniti Research](#) indicated that the number of cloud gaming users would continue to grow rapidly by 29% during the period of 2021. The same report also indicated that in terms of geography, the Americas account for the maximum market share in 2016 due to the high penetration of smart devices and the availability of high-speed internet in the region. However, the APAC region is also expected to grow significantly during the forecast period as developing countries, including China, South Korea, and India, are rapidly adopting novel technologies for gaming [[Infiniti Research 2016](#)].

The increasing popularity of cloud gaming can be attributed to several advantages to gamers, game developers, and service providers. For gamers, cloud gaming

enables them to: (i) have access to their games anywhere and anytime, (ii) purchase or rent games on-demand, (iii) avoid regularly upgrading their hardware, and (iv) enjoy unique features such as migrating across client computers during a game session, observing ongoing tournaments, and sharing game replays with friends. For game developers, cloud gaming allows them to: (i) concentrate on a single platform, which in turn reduces the porting and testing efforts, (ii) bypass retailers for higher profit margins, (iii) reach out to more potential gamers with a rental rather than a purchase model, and (iv) avoid piracy as the game software is never downloaded to client computers. For service providers, cloud gaming: (i) leads to new business models, (ii) creates more demands on already-deployed cloud resources, and (iii) demonstrates the potentials of new remote execution applications, as cloud gaming requires strict constraints on computing and networking resources.

Despite the great opportunities of cloud gaming, several crucial challenges must be addressed before it reaches its full potential to engage more gamers, game developers, and service providers. First, cloud gaming platforms and testbeds must be built up for comprehensive performance evaluations. The evaluations include measurements on Quality of Service (QoS) metrics, such as energy consumption and network metrics, and Quality of Experience (QoE) metrics, such as gamer-perceived gaming experience. Building platforms and testbeds, designing test scenarios, and carrying out performance evaluations require significant efforts, while analyzing the complex interplay between QoS and QoE metrics is even more challenging. Second, the resulting platforms and evaluation procedures allow the research community to optimize the system components, such as cloud servers and communication channels. Optimization techniques for better resource allocation are strongly demanded for cloud servers, where better content coding and adaptive transmissions are also important in order to further improve the overall experience and resource efficiency of cloud gaming services. Thus, many research papers have been published in order to address various technical challenges toward the realization of an ideal cloud gaming system. Figure 11.1 shows a typical architecture for a cloud gaming system.

In this chapter, we first provide an overview of cloud gaming research developed in the last decade in Section 11.1. We classify these papers into four categories: platforms, cloud deployment, client design, and communications, and then present each category of research in Sections 11.2, 11.3, 11.4, and 11.5, respectively. Finally, we depict our projections on the future paradigms of cloud gaming services in Section 11.6 and conclude in Section 11.7.

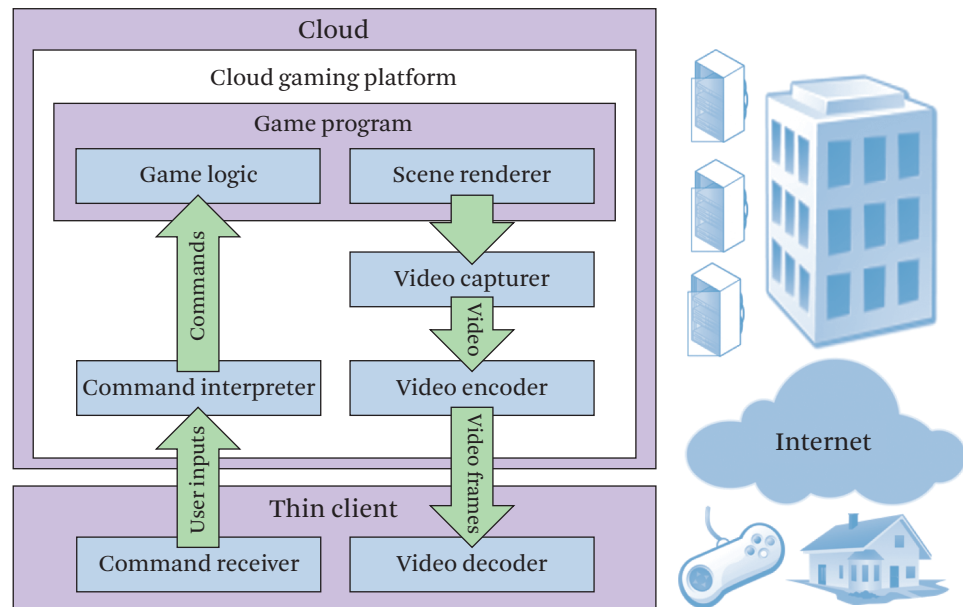


Figure 11.1 Illustration of a typical cloud gaming system architecture. (Adapted from Cai et al. [2016b])

11.1 Overview on Cloud Gaming Research

As a promising cloud service provisioning paradigm, cloud gaming has attracted interest from a number of research groups all over the world. These teams have shared their thoughts and ideas in this area from their viewpoints. In this section, we survey and summarize all of these cloud gaming overviews.

Ross [2009] was the first paper to introduce the cloud gaming model to academia in 2009, nine years after the G-cluster's demonstration of a prototype cloud gaming technology at E3. In this article, the author describes gaming as a killer application of cloud computing and depicts the blueprint of this novel game delivery paradigm, proposed by Advanced Micro Devices (AMD), that renders games' scene videos, compresses them, and transmits them to gamers through the Internet. This approach enables computer gamers to offload their graphic rendering tasks to the cloud, and thus eliminates the computation workload on the gamers' side. This is the most popular definition of cloud gaming adopted by most of the research work in this area. However, a 2014 publication Mishra et al. [2014] provides a more general definition, by envisioning the cloud gaming system as a novel computer

architecture that leverages cloud resources to improve gaming performance, such as rendering quality, response time, precision, and fairness. The authors distribute system workload to multiple cloud servers and game clients to realize this vision. For a further step, [Cai et al. \[2014\]](#) explore the essence of cloud games as interdependent components, and thus define cloud gaming as utilizing cloud resources to host gaming components, thereby reducing workload at the terminal and increasing the overall system performance. Citing different integration approaches of the cloud, the authors identify and discuss the research directions of three cloud gaming architectural frameworks, which are Remote Rendering, Local Rendering, and Cognitive Resource Allocation.

After the official launch of OnLive in March 2010, the business model for cloud gaming became a hot topic in both the industry and academia communities. In their 2011 article, [Riungu-Kalliosaari et al.](#) conduct interviews in small and medium size game companies to study the adoption dynamics of cloud computing, in a qualitative approach. With the grounded theory method, the authors observe that although the concept of cloud gaming is relatively well known in the industry, game developers are still hesitant to adopt cloud computing services and technologies due to the lack of clear business models and success stories. To this end, [Ojala and Tyrvaainen \[2011\]](#) begin their investigations on developing business models for cloud gaming services. As a case study for Software as a Service (SaaS), the authors select G-cluster, one of most famous cloud gaming companies, to compare its business model over five years from 2005 to 2010. They conclude that the business model in cloud gaming has become simpler and has fewer actors, which increases the revenue per gamer. In addition, they anticipate that the cloud gaming technology will render copyright infringement practically impossible. Meanwhile, another paper [\[Moreno et al. 2012\]](#) considers the convergence of mobile and cloud in the game industry from a business model perspective. The authors discuss the first sketch of a possible business model of the Kusanagi project, a proposed end-to-end infrastructure, from domains of service, technology, organization, and finance, and compare three examples over these domains, including G-Cluster, Gaikai, and OnLive.

On the other hand, during the past decade, there have been numerous cloud gaming systems and services in the market. A number of studies examine these systems and envision the opportunities, challenges, and directions in this area. These papers, e.g., [\[Dey et al. 2012\]](#), [\[Soliman et al. 2013\]](#), [\[Wu 2014\]](#), [\[Cai et al. 2014\]](#), [\[Mishra et al. 2014\]](#), [\[Chen et al. 2014\]](#), [\[Chuan et al. 2014\]](#), cover most of the commercial and open platforms, while their concerns about technical issues are greatly overlapped by the topics of response time minimization, graphical

video encoding, network-aware adaption, QoE optimization, and cloud resource management.

Besides the common focuses, each research group has particular interests and directions. For example, [Dey et al. \[2012\]](#) concentrate on developing device-aware scalable applications, which involve an open issue of how to extend the cloud to wireless networks. In [Soliman et al. \[2013\]](#), the authors discuss legal issues, including patents, ownership concerns, guaranteed service levels, and pricing schemes. In contrast, piracy and hacking may no longer be issues, since the executable game program will not be delivered to the gamers. On the other hand, [Wu \[2014\]](#) explores the architecture of cloud gaming systems from the aspect of cloud service layers, namely IaaS, SaaS, and PaaS, where security is identified as a great potential challenge in cloud gaming, especially data protection and location. [Cai et al. \[2014\]](#) investigate the features of different game genres and identify their impact on cloud gaming system design. In addition, they provide a vision on GaaS (i.e., Games as a Service) provisioning for mobile devices. [Mishra et al. \[2014\]](#) explain how to enhance the quality of online gaming by integrating several techniques, including the interplay between QoS and QoE metrics, game models, and cloud expansion. [Chen et al. \[2014\]](#) point out a number of distinct research directions in cloud gaming, such as game integration, visualization, user interface, server selection, and resource scheduling. [Chuan et al. \[2014\]](#) study cloud gaming from a green media perspective. The authors discuss the major cloud gaming subsystems with green designs, which include a cloud data center, graphics rendering module, video compression techniques, and network delivery schemes.

In addition to the above general studies, there exists a large amount of cloud gaming research that focuses on individual research problems. Therefore, we refer the readers to [Cai et al. \[2016b\]](#) for a more comprehensive literature survey in the cloud gaming field.

11.2 GamingAnywhere: An Open-Source Cloud Gaming Platform

Remote desktop software packages, such as [LogMeIn](#), [TeamViewer](#), and [UltraVNC](#), have been popular for some time, but were not designed for highly interactive applications, and thus do not meet the strict requirements of cloud gaming [[Chang et al. 2011](#)]. Although there exist commercial cloud gaming services, e.g., [GaiKai](#), [OnLive](#), and [StreamMyGame](#), in a 2011 measurement study [Chen et al. \[2011\]](#) report that these cloud gaming systems still suffer from high response delay, among other limitations. For example, assuming a negligible network latency, 134 ms and 375 ms mean response delay are measured on the OnLive and StreamMyGame

platforms, respectively [Chen et al. 2011]. Hence, the challenges of developing cloud gaming systems for high video quality and low response delay remain open. We consider a major cause of the unsatisfactory performance of existing cloud gaming systems to be the lack of an *open-source* cloud gaming system, which would enable the research community to readily implement and evaluate their new ideas for better designs of cloud gaming systems.

In this section, we present a guide to *GamingAnywhere*, the first and probably the most widely adopted open-source cloud gaming system, which is now publicly available on <http://gaminganywhere.org/> [GamingAnywhere Repository 2013]. This system has three main advantages over many existing systems.

1. GamingAnywhere is an *open* system, in the sense that a component of the video streaming pipeline can be easily replaced by another component implemented with a different algorithm, standard, or protocol. For example, GamingAnywhere by default uses x264 [X264 2012] and vpxenc [WebM 2013] to encode the captured raw gameplay videos. To expand GamingAnywhere for stereoscopic games, an H.264/MVC encoder may be plugged into it without significant changes. More generally, various algorithms, standards, protocols, and system parameters can be rigorously evaluated with real experiments, which is difficult, if not impossible, on proprietary cloud gaming systems.
2. GamingAnywhere is cross-platform, and is currently available on Windows, Linux, OS X, and Android (client only). This is made possible largely due to the modularized design of GamingAnywhere.
3. GamingAnywhere has been designed to be efficient, for example, in its minimizing of time and space complexity by using shared circular buffers to reduce the number of memory copy operations. These optimizations allow GamingAnywhere to provide a high-quality gaming experience with short response delays. In particular, on a commodity Intel i7 PC, GamingAnywhere delivers real-time 720p videos at ≥ 35 fps, which is equivalent to a processing time of less than 28.6 ms for each video frame, with a video quality significantly higher than that of some commercial cloud gaming systems. In particular, GamingAnywhere outperforms OnLive by 5 dB in Peak Signal-to-Noise Ratio (PSNR) according to Huang et al. [2014a].

11.2.1 Target Users

GamingAnywhere is designed for three types of users. First, researchers and engineers may use GamingAnywhere to evaluate their new ideas without reimplementing the irrelevant software components. This will allow many new ideas to be tested

in real testbeds and deployed in practical applications within short time frames. Second, cloud gaming service providers may develop a cloud gaming front-end for gamers to choose the desired games, and integrate their front-end with GamingAnywhere. Compared to developing a brand new cloud gaming system from scratch, adopting GamingAnywhere would largely reduce the time to market of new services. Third, gamers may set up GamingAnywhere on their powerful home PCs, and enjoy high-quality gaming experiences using thin clients via a LAN or the Internet.

To reach a wider audience, GamingAnywhere is written in C/C++ and it leverages several popular open-source libraries such as *ffmpeg* and *libSDL*. Detailed designs and implementations can be found in Huang et al. [2014a]. Users are also encouraged to contribute to this project by installing and working with the system, providing comments and suggestions, sharing game configurations, and even submitting patches via the forum or public repositories.

11.2.2 System Architecture

The game selected by a user runs on a game server. There is an agent running along with the selected game on the same server. The agent can be a stand-alone process or a module (in the form of shared object or dynamically-linked libraries (DLL)) injected into the selected game process. The choice depends on the type of game and how the game is implemented. The agent has two major tasks: The first task is to capture the A/V frames of the game, encode the frames using the chosen codecs, and then deliver the encoded frames to the client via the data flow. The second task of the agent is to interact with the game. Whenever it receives the user's actions transmitted from the client, it injects the received keyboard, mouse, joystick, and even gesture events into the game process, which will be then translated to in-game actions. However, as there exist no standard protocols for delivering users' actions, we design and implement the transport protocol of user actions in GamingAnywhere.

Technically, the client program of GamingAnywhere is implemented by combining an RTSP/RTP multimedia player and a keyboard/mouse logger. GamingAnywhere allows *observers*¹ by nature because the server delivers encoded A/V frames using the standard real time streaming protocol (RTSP) and real time transport protocol (RTP) protocols. In this way, an observer can watch gameplay by simply accessing the corresponding game URL with full-featured multimedia players, such as the VLC media player [VideoLAN 2017], which are available on almost all OSs and platforms.

1. In addition to playing a game themselves, hobbyists may like to watch how other gamers play the game. An observer can only watch how a game is played but cannot be involved during gameplay.

GamingAnywhere defines two types of network flows in the architecture, the data flow and the control flow. Whereas the data flow is used to stream audio and video (A/V) frames from the server to the client, the control flow runs in a reverse direction, being used to send the user's actions from the client to the server. This architecture of GamingAnywhere allows it to support both PC- and Web-based games.

11.2.3 Environment Setup

GamingAnywhere is released with two types of software packs: `all-in-one` and pre-compiled binary packs [GamingAnywhere Repository 2013]. The `all-in-one` pack includes source codes, third-party library source code, and pre-compiled binaries, while a pre-compiled binary pack such as the `bin.win32` pack includes only pre-compiled binaries for running the software on a certain platform. In any case, users can always build their own binaries that best match their environments.

GamingAnywhere can be installed by simply uncompressing the software packs, with or without the source code. If the `all-in-one` software pack is downloaded, the complete GamingAnywhere system can be compiled from scratch. One exception is the dependent Windows libraries: as re-building them can be tricky, the pre-built Win32 libraries are included in the software packs.

We briefly describe how to build GamingAnywhere binaries below. GamingAnywhere consists of three binaries: (i) `ga-client`, which is the thin client, (ii) `ga-server-periodic`, which periodically captures the desktop or a window as the game screen, and (iii) `ga-server-event-driven`, which directly hooks into the game executables to capture game screens.

POSIX. To build GamingAnywhere on POSIX platforms (i.e., Linux and Mac OS X), `g++`, `pkg-config`, `libX11`, `libXext`, `libXtst`, `libfreetype6`, and `libasound2` are required. In addition, `libgl1-mesa`, `libglu1-mesa`, and `libpulse` need to be installed. You must install both binaries and development files for the above packages. Next, the following instructions are recommended:

1. Edit the `env-setup` script and point `GADEPS` to the absolute path of `gaming-anywhere/deps.posix/`.
2. Load the environment variables from `env-setup` by using the `.` or `source` commands.
3. Build file dependencies using `make` under `deps.src/`.
4. Build GamingAnywhere using `make all` under `ga/`.
5. Install GamingAnywhere using `make install` under `ga/`. All the generated binaries will be copied into `bin/`.

Windows. To build GamingAnywhere on Microsoft Windows from the source code, Visual Studio 2010, Windows SDK, and DirectX SDK are required. Then, follow the instructions:

1. Install dependencies libraries by running `install.cmd` under `deps.pkg.win32\`.
2. Install DirectX SDK into `C:\Microsoft DirectX SDK`. If you install it at a different location, modify the SDK paths in `ga\module\vsource-desktop\NMakefile.d3d` and `ga\server\event-driven\NMakefile`.
3. Launch Microsoft Visual C++ command line prompt.
4. Build GamingAnywhere by running `nmake /f NMakefile all` command under `ga\`.
5. Install GamingAnywhere by running `nmake /f NMakefile install` command under `ga\`. All the generated binaries will be copied into `bin\`.

11.2.4 Execution

Before executing GamingAnywhere, please make sure the following prerequisites are met. On POSIX, add `deps.posix/lib` to the system-wide `ld.so` search path. Moreover, modify `bin/config/common/server-common.conf` to ensure that `DISPLAY` is set to an X-Window desktop with the `XTEST` extension enabled. The X-Window desktop can be a VNC or `Xvfb` desktop. On Windows, copy `deps.win32\bin*.dll` to the same directory of GamingAnywhere binaries (i.e., `bin\`) or a directory in the system-wide DLL search path.

The examples below describe how to run the GamingAnywhere server and client using DirectX SDK samples on Windows. Certainly it is possible to modify and run GamingAnywhere in many ways beyond these examples.

Start up the periodic capture server. To stream a game window to the client, launch the game first and then use the following command: `ga-server-periodic config\server.d3dex.conf`. The server configuration file sets either the `find-window-name` or `find-window-class` option to specify the title and the class name, respectively, of the game window to be streamed. Then, `ga-server-periodic` moves the specified window to the upper-left corner of the desktop, and starts capturing the screens of the window periodically (say, every 20 ms). Gamers should avoid moving the window manually as it may affect the mouse cursor functionality. If you do not know the window title or the class name, you can find the information for a window by using the `Spy++` tool and the `xwininfo` tool on Windows and Linux, respectively.

Start up the event-driven server. Rather than capturing a game’s window periodically, GamingAnywhere provides another type of built-in server that is able to capture the game window whenever the game updates its screen via API hooking. Please issue the following command: `ga-server-event-driven config\server.d3dex.conf`. (Because `ga-server-event-driven` launches the game executable specified by the `game-dir` and `game-exe` options in the configuration file, each game has its own configuration file.)

Launch the client. The GamingAnywhere thin client can be run by `ga-client config\client.abs.conf rtsp://192.168.1.1:8554/desktop`, where the server IP address is 192.168.1.1; by default GamingAnywhere uses TCP port 8554 for RTSP streams and TCP/UDP port 8555 for control messages. The `client.abs.conf` is a sample configuration that enables the absolute mouse cursor positioning. In contrast, `client.rel.conf` employs the relative mouse cursor positioning. The choice of either configuration file is game-dependent, as different games use different mechanisms for mouse position acquisition and management.

Figure 11.2 shows a running GamingAnywhere testbed. In this picture, the laptop on the left-hand side runs the *LEGO Batman* game with the GamingAnywhere event-driven server (i.e., `ga-server-event-driven`). The MacBook laptop on the right-hand side runs the GamingAnywhere client. Two observer clients also connect to the same server simultaneously. One client runs on an Android phone and another client runs on an iPad 2. Moreover, as the GamingAnywhere server delivers audio and video frames using the standard RTSP/RTP protocol, any compatible RTSP/RTP players can be used to observe gameplay.

We provide configuration files for several public-domain games [[GamingAnywhere Repository 2013](#)], such as *Armagetron Advanced*, *Assault Cube*, and *Neverball*, so that anyone can quickly experience GamingAnywhere. Notice that the server configuration files are game-dependent, and therefore we provide a sample server configuration for each game. To get the server configuration working, please modify the `game-dir` and `game-exe` options in the configuration files to point to the game executables.

11.2.5 Research Based on GamingAnywhere

GamingAnywhere has served the basis of numerous research studies in academia and the industry. For example, [Hong et al. \[2014a\]](#) utilized GamingAnywhere in their study of how to allocate virtual machines inside and across data centers in order to co-optimize gamer QoE and provider net profit. [Slivar et al. \[2014\]](#) used GamingAnywhere in their in-home streaming scenario to compare the QoE



Figure 11.2 Demonstration of a running GamingAnywhere system. There are four devices in the photo: one game server (the left-hand side laptop) and three game clients (a MacBook, an Android phone, and an iPad 2). (Adapted from Huang et al. [2013a]) (LEGO Batman software is copyrighted by Traveller's Tales Games Publishing Ltd.)

differences between online gaming and cloud gaming, and explore the effect of various factors, such as network conditions and player skills, on the perceived QoE levels. Hsu et al. [2015] regarded GamingAnywhere as a general-purpose video-based screencast platform and took advantage of its customizability to investigate the impact of individual components, such as GPU and transport protocol, on the overall performance in a formal comparison of screencast solutions. Besides all these, there has been much more research work done using GamingAnywhere in the studies of mobile cloud gaming, e.g., [Huang et al. 2013b], GPU consolidation, e.g., [Hong et al. 2014b], energy consumption, e.g., [Huang et al. 2014b], and so on.

Readers, certainly, are encouraged to conduct their own research based on GamingAnywhere by exploiting its high extensibility, portability, configurability, and openness.

11.2.6 Community Participation

GamingAnywhere provides an interactive web forum for discussions among developers, researchers, and users at <http://gaminganywhere.org/forum>. The forum provides several advantages over email communications, such as: 1) the questions and responses are posted on the forum so that users can be pointed to a certain

thread whenever applicable; 2) enthusiastic users often appear and help each other to resolve various issues on the installation and execution of GamingAnywhere.

As some researchers and users are keen to contribute their improvements and add-ons back to the GamingAnywhere project, the full source code of GamingAnywhere is also hosted on a github repository to enable a wider and more active participation in future development. Readers are very much welcome and encouraged to be part of this non-profit and open-source project.

11.3 Cloud Deployment

Fueled by elastic resource provisioning, reduced costs, and unparalleled scalability, *cloud computing* is drastically changing the operation and business models of the IT industry. Advances in cloud technology have expanded to facilitate offloading more complex tasks as high-definition 3D rendering, which turns the idea of cloud gaming hosted in a public cloud environment into a reality.

Originally, cloud gaming platforms tended to focus on private, non-virtualized environments with proprietary hardware, where each user is often mapped in a one-to-one fashion to a physical machine in the cloud. Modern public cloud platforms heavily rely on *virtualization*, which allows multiple *virtual machines* to share the underlying physical resources, making truly scalable *play-as-you-go* service possible. Despite the simplicity and ease of deployment, existing cloud gaming platforms have seldom been deployed in the public cloud environment.

Migrating gaming to a public cloud (e.g., Amazon EC2) is non-trivial, however. The system modules should be carefully planned for effective virtual resource sharing with minimum overhead. Moreover, as the complexity of 3D rendering increases, modern game engines not only rely on the general-purpose CPU for computation, but also on dedicated *graphical processing units* (GPUs). While GPU cards have been virtualized to some degree in modern virtualization systems, their performance has historically been poor given the ultra-high memory transfer demand and the unique data flows. Recent advances in terms of both hardware and software design have not only increased the usability and performance of GPUs, but created a new class of GPUs specifically for virtualized environments. A representative is NVIDIA's GRID Class GPUs, which allow several virtualized systems the ability to each utilize a dedicated GPU, by placing several logical GPUs on the same physical GPU board. Hardware advances greatly assist in the deployment of online gaming systems in a public cloud environment. In this chapter we will discuss some of the technologies and issues inherent in bridging the cloud gaming systems and the public cloud.

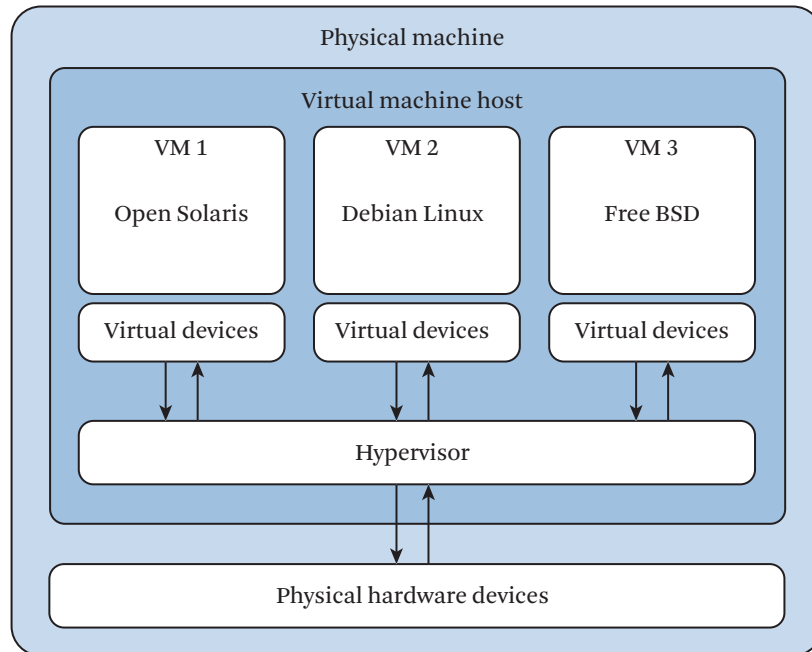


Figure 11.3 Hypervisor-based computer virtualization. (Adapted from [Shea et al. \[2015\]](#))

11.3.1 Virtualization: A Recap

Any virtualization system must ensure that each virtual machine (VM) be given fair and secure access to the underlying hardware. In state-of-the-art virtualization systems, this is often achieved through the use of a software module known as a *hypervisor*, which works as an arbiter between a VM's virtual devices and the underlying physical devices. The use of a hypervisor brings many advantages, such as device sharing, performance isolation, and security between running VMs. However, having to consult the hypervisor each time a VM makes a privileged call introduces considerable overhead as the hypervisor must be brought online to process each request. Figure 11.3 shows the interactions between three different VMs and the hypervisor.

A GPU consists of hundreds or even thousands of cores, allowing a vast amount of threads to run simultaneously to solve computationally intensive tasks. Due to its intrinsically parallel nature, a GPU demands much higher memory bandwidth than a CPU, and thus existing hardware designs around GPUs are mostly *throughput-driven*. For instance, the GPU's internal bandwidth is optimized by using a dedicated memory, a newer generation of which is GDDR5. Externally, a GPU

is interfaced with the motherboard by a PCI Express (PCI-E) expansion slot, which has a much higher bus throughput than other older PCI buses. Unfortunately, the data transfer between the main memory and GPU is still a bottleneck, and virtualization can dramatically degrade the memory transfer performance. Using SIMD (single instruction, multiple data), a GPU is able to unleash its computing power on data-parallel computations. However, this makes it harder to share a GPU among multiple VMs, each with its own distinct task to perform. The CPU, on the other hand, excels at task-parallel computations with its massive number of cores, making virtualization easier with concurrent access to different VMs and their disparate tasks.

11.3.2 Virtualized GPU Architecture and Pass-through

Recent hardware advances have enabled virtualization systems to perform a one-to-one mapping between a device and a virtual machine guest, allowing hardware devices that do not virtualize well to still be used by a VM, including a GPU. Both Intel and AMD have created hardware extensions for such device *pass-through*, namely VT-D by Intel and AMD-Vi by AMD. They work by making the processor's *input/output memory management unit* (IOMMU) configurable, allowing the system's hypervisor to reconfigure the interrupts and *direct memory access* (DMA) channels of a physical device, to map them directly into one of the guests.

As illustrated in Figure 11.4(a), data flows through DMA channels from the physical device into the memory space of the VM host. The hypervisor then forwards the data to a virtual device belonging to the guest VM. The virtual device interacts with the driver residing in the VM to deliver the data to the guest's virtual memory space. Notifications are sent via interrupts and follow a similar path. Figure 11.4(b) shows how a 1-1 device pass-through to a VM is achieved. As can be seen, the DMA channel can allow data to flow directly from the physical device to the VM's memory space. Also, interrupts can be directly mapped into the VM through the use of remapping hardware, which the hypervisor configures for the guest VM.

The advanced pass-through grants a single VM a one-to-one hardware mapping between itself and the GPU. These advances have allowed the cloud platforms to offer virtual machine instances with GPU capabilities. For example, Amazon EC2 has added an instance class known as the GPU Instances, which have dedicated NVIDIA GPUs for graphics and general-purpose GPU computing.

Although many cloud computing workloads do not require a GPU, cloud gaming servers require access to a rendering device to provide 3D graphics. As such, VM and workload placements have been researched to ensure cloud gaming servers have access to adequate GPU resources. In 2011, Kim proposed a novel architecture to

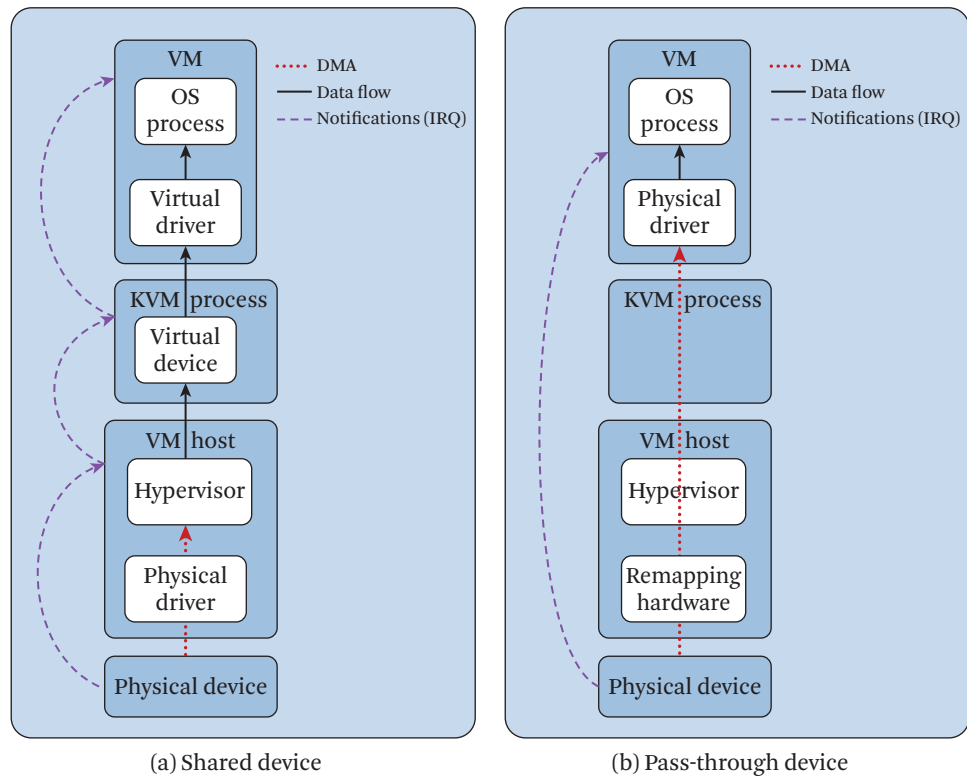


Figure 11.4 Shared vs pass-through device simplified architecture. (Adapted from [Shea et al. \[2015\]](#))

support multiple-view cloud gaming servers, which share a single GPU. This architecture provides multi-focal points inside a shared cloud game, allowing multiple users to potentially share a game world, which is rendered on a single GPU. In 2012, researchers at the University of Southern California performed an analysis of the performance of combined CPU/GPU servers for game cloud deployments [[Zhao et al. 2012](#)]. The researchers tested offloading different aspects of game processing to these cloud servers, while maintaining some local processing at the client side. They concluded that leaving some processing at the client side could lead to an increase in QoS of cloud gaming systems.

In 2013 it was shown that direct GPU assignment to a virtualized gaming instance can lead to frame rate degradation of over 50% in some gaming applications [[Shea et al. 2013](#)]. The researchers found that the GPU device pass-through severely diminished the data transfer rate between the main memory and the GPU. In 2015, the work was extended to more advanced platforms and found that although the

memory transfer degradation still existed, it no longer affected the frame rate of current-generation games [Shea et al. 2015]. Hong et al. [2014b] performed parallel work, where the researchers discovered that the frame rate issue present in virtualized clouds may be mitigated by using mediated pass-through, instead of direct assignment. Work has also been done on encoding for applications such as cloud gaming.

11.4 Thin Client Design

From live video playback to low-latency networking, a cloud gaming thin client relies on a number of cutting-edge technologies to function. We begin our analysis with the important design considerations that must be addressed by a cloud gaming system. A cloud gaming thin client system must collect a player's actions, transmit them to the cloud server where the actions are processed, and then transmit the data back to the thin client and decode it. To ensure interactivity, these actions must happen in the order of milliseconds. Intuitively, this amount of time, which is defined as *interaction delay*, must be kept as short as possible in order to provide a rich experience to the cloud game players. However, there are trade-offs: the shorter the player's tolerance for interaction delay, the less time the system has to perform such critical operations as scene rendering and video compression. Also, the lower this time threshold is, the more likely a higher network latency can negatively affect a player's experience of interaction. With this in mind, we will now discuss delay tolerance.

11.4.1 Interaction Delay Tolerance

Research on traditional gaming systems has found that different styles of games have different thresholds for maximum tolerable delay [Claypool and Claypool 2006]. Table 11.1 summarizes the maximum delay that an average player can tolerate before the QoE begins to degrade. As a general rule, games that are played in the first-person perspective, such as the shooter game *Counter-Strike*, become noticeably less playable when actions are delayed by as little as 100 ms. This low delay tolerance is because such first-person games tend to be action-based, and players with a higher delay tend to have a disadvantage. In particular, the outcome of definitive game-changing actions such as who “pulled the trigger” first can be extremely sensitive to the delay in an action-based first-person shooter (FPS) game. Third-person games, such as role playing games (RPGs), and many massively multiplayer games, can often have a higher delay tolerance of up to 500 ms. This is because a player's commands in such games—for example, use an item, cast a spell, or heal a character—are generally executed by the player's avatar; there is often an invocation

Table 11.1 Delay tolerance in traditional gaming

Example Game Type	Perspective	Delay Threshold
First person shooter (FPS)	First-person	100 ms
Role playing game (RPG)	Third-person	500 ms
Real time strategy (RTS)	Omnipresent	1000 ms

phase, such as chanting magic words before a spell is cast, and therefore the player does not expect the action to be instantaneous. The actions must still be registered in a timely manner, since the player can become frustrated if the interaction delay causes them a negative outcome—for example, they healed before an enemy attack but still died because their commands were not registered by the game server due to high latency. The last category of games are those played in an “omnipresent” view, that is, a top-down view looking at many controllable entities. Examples are real time strategy (RTS) games like *Star-Craft* and simulation games such as *The Sims*. Delays of up to 1000 ms can be acceptable to these styles of games since the player often controls many entities and issues many individual commands, which often take seconds or even minutes to complete. In a typical RTS game, a delay of up to 1000 ms for a build-unit action that takes over a minute can often be unnoticed by the casual player.

Although there is some similarity between interaction delay tolerance for traditional gaming and cloud gaming, we must note the following critical distinctions. First, traditionally, the interaction delay was only an issue for multiplayer online gaming systems, and was generally not considered for single-player games. Cloud gaming drastically changes this; now all games are being rendered remotely and streamed back to the player’s thin client. As such, we must be concerned with interaction delay even for a single-player game. Also, traditional online gaming systems often hide the effects of interaction delay by rendering the action on a player’s local system before it ever reaches the gaming server. For example, a player may instruct the avatar to move and it immediately begins the movement locally; however, the gaming server may not receive the update on the position for several milliseconds. Since cloud gaming offloads its rendering to the cloud, the thin client no longer has the ability to hide the interaction delay from the player. Visual cues such as mouse cursor movement can be delayed by up to 1000 ms, making it impractical to expect the player will be able to tolerate the same interaction delays in cloud gaming as they do in traditional gaming systems. The maximum interaction delay for all games hosted in a cloud gaming context should be at most 200 ms. Other games, specifically such action-based games as first-person shooters, likely require less than 100 ms interaction delay in order not to affect the player’s QoE.

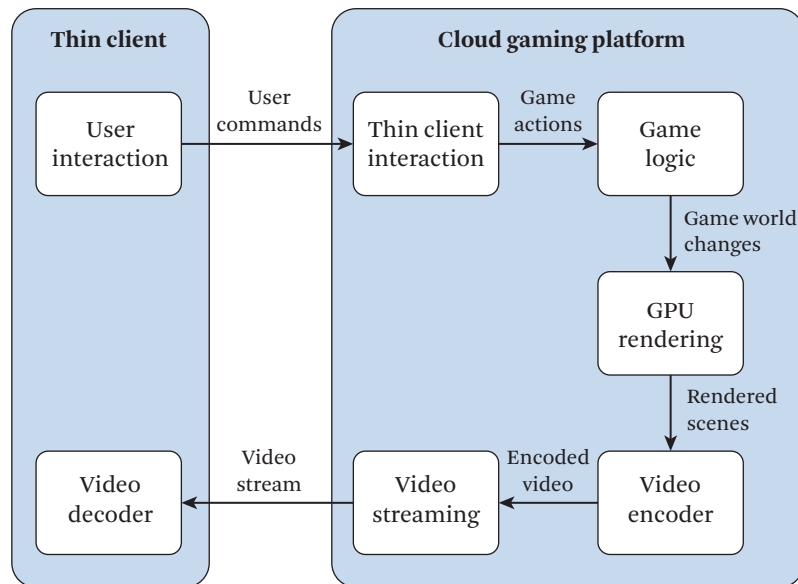


Figure 11.5 Thin client commands and data flow. (Adapted from Cai et al. [2016a])

Based on the design considerations we have been discussing, a generic architecture and data flow of a cloud gaming thin client can be seen in Figure 11.5. As can be observed, a player’s commands must be sent over the Internet from its thin client to the cloud gaming platform. Once the commands reach the cloud gaming platform, they are converted into appropriate in-game actions, which are interpreted by the game logic into changes in the game world. The game world changes are then processed by the cloud system’s graphical processing unit (GPU) into a rendered scene. The rendered scene must be compressed by the video encoder, and then sent to a video streaming module, which delivers the video stream back to the thin client. Finally, the thin client decodes the video and displays the video frames to the player.

11.4.2 Real-World Thin Client: Configuration and Performance

A key motivation of migrating games to the cloud is to enable resource-constrained (in terms of computation, memory, battery, etc.) thin clients to play advanced games. Hence, we now shift our focus to the configuration and performance of a thin client in our system.

Our test system was an EVGA Tegra NOTE 7 tablet, powered by NVIDIA’s set Tegra 4, functionally a SoC (System on Chip) that features a quad-core ARM Cortex-

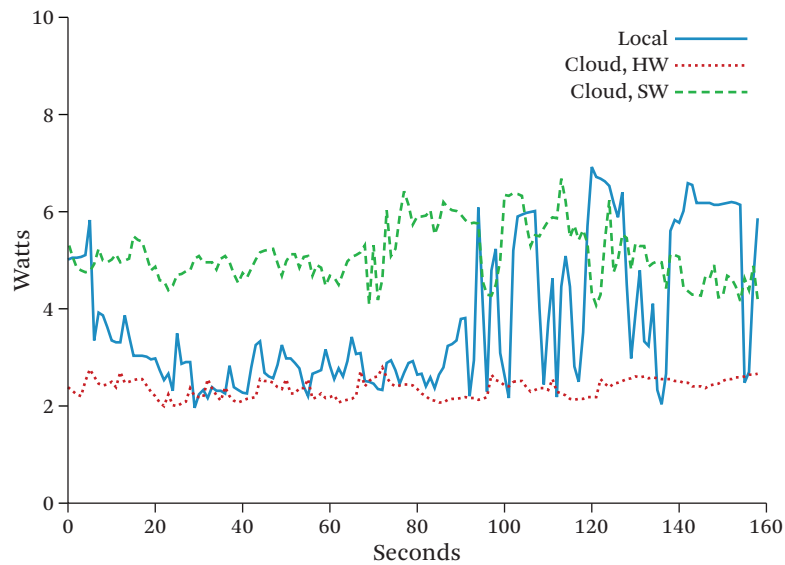


Figure 11.6 Ice storm power consumption. (Adapted from Shea et al. [2015])

A15 CPU, a 72-core NVIDIA GeForce GPU, and particularly a dedicated video decoding block that provides full support for hardware decoding. The tablet has a 1280×800 (720p) native resolution screen; yet the GPU can support up to 4K UHD external display through its video output. The installed Android 4.2.2 Jelly Bean enables users to explore and utilize the device with a wide range of gaming apps.

We configured our Tegra tablet to make use of its built-in hardware decoder and to switch between hardware decoding and software decoding.

Hardware vs. software video decoders: Power consumption. We use an advanced 3D benchmark for mainstream tablets, namely, 3DMark Ice Storm Benchmark, which includes 720p game scenes, two graphics tests, and a physics test to stress GPU and CPU, respectively. We installed it not only on the Tegra tablet for the power measurement of local rendering, but also on the cloud gaming server for remote rendering.

In Figure 11.6, we depict the results of the *3DMark Ice Storm*. The horizontal axis represents the timeline of the benchmark in seconds, and the vertical denotes the overall power consumption in watts. On the bottom sits the hardware decoding (average 2.36 watts), which stayed stable throughout the entire benchmarking. The local rendering is represented as the solid line sitting roughly in the middle (average 3.75 watts), which started high at around 5 watts, dropped under 4 and

remained there until the later part of the benchmark. The other line sitting roughly on the top (average 5.11 watts) gives the result with software decoding. As we can see, when applying the hardware decoding, the device had an overall power consumption between 2 and 3 watts, while local rendering stayed competitive with the hardware decoding in the middle of benchmarking, where only normal game scenes were rendered and no benchmark loading or other ultra CPU-intensive works were happening. The software decoding, however, doubled this amount to over 5 watts, which was even worse than the local rendering in terms of supporting usual game scenes.

11.4.3 Hardware Decoder: Key for Cloud Gaming Client

A deeper investigation shows that local rendering increases the power consumption at the very beginning and particularly in the later part of the benchmark. We believe that this occurs when a large amount of CPU and GPU calls were made to load and initialize the benchmark, stressing the device in the graphics and physics test. This conjecture is not only based on our track of CPU/GPU usage, but also on the fact that the sharp increase of power consumption, on the timeline, compactly corresponds to the benchmark loading and stress testing, where extraordinarily intensive workloads were involved. In other words, as opposed to traditional local-rendered gaming, cloud gaming saves not only a large amount of power consumed in rendering game scenes, but in game loading and other CPU/GPU-intensive operations as well.

The savings, however, comes from hardware decoding only. The software decoder on the client side consumes even more energy than local rendering, by 5.11 watts against 3.75, theoretically cutting down 27% of the battery life, and making cloud gaming less appealing given the notorious shortage of power supply in mobile devices. The hardware decoder, on the other hand, holds a much lower average power consumption at 2.36 watts, impressively extending battery life by 59% longer than with local rendering and 117% longer than by the software decoder, which makes battery shortage much less of a concern for users.

Moreover, as exhibited in the video clip experiment, when video resolution increased from 480p to 720p and to 1080p, the software decoder expended significantly more computing overhead and power consumption, while the hardware decoder managed to contain the computational and energy cost to a relatively low amount. Given that mainstream tablets now stand at 720p resolution and are still moving forward, the gap between software and hardware decoders will only increase. For richer and more detailed game world rendering at 4K UHD resolution and beyond, a hardware decoder is definitely needed, though it remains to be universally supported by tablets and smartphones.

11.5 Communication

In this section, we discuss two research directions on the aspect of communication between a cloud gaming client and its server. The main challenges lie in the demand for high-definition game scenes rendered on game servers to be presented on game clients at an update frequency of 60 Hz or even higher, which introduces a high communication need. Therefore, we first discuss three common strategies (which can be used simultaneously) for reducing the communication workload. Secondly, we give an overview of the state-of-the-art research on how to make the communication adaptive to the dynamic Internet bandwidth in order not to cause network congestion while maintaining high data transmission efficiency.

11.5.1 Data Compression on Communications

After game scenes are computed on cloud servers, they have to be captured in proper representations and compressed before being streamed over networks. Beyond the conventional real-time video compression techniques that are widely applied to video-on-demand services, compression approaches for cloud gaming can be categorized into three schemes: (i) *video compression*, which encodes two-dimensional (2D) rendered videos and potentially auxiliary videos (such as depth videos) for client-side post-rendering operations, (ii) *graphics compression*, which encodes 3D structures and 2D textures, and (iii) *hybrid compression*, which combines both video and graphics compression.

Video compression utilizes graphics contexts to reduce the server transmission rate. The work [Shi et al. 2011] introduces a video encoder that selects a set of key frames in the video sequence and uses the 3D image warping coding to interpolate other non-key frames. This approach takes advantage of the pixel depth, rendering viewpoints, camera motion patterns, and even the auxiliary frames that do not actually exist in the video sequence to assist video coding. Another work [Xu et al. 2014] rectifies the camera rotation to produce video frames that are more motion-estimation friendly. On client computers, the rectified videos are compensated with some camera parameters using a lightweight 2D process. In addition, a new interpolation algorithm is designed to preserve sharp edges, which are common in game scenes.

Graphics compression is proposed for better scalability, because 3D rendering is done on individual client computers. Compressing graphics data, however, is quite challenging and may consume excessive network bandwidth. Lin et al. [2014] designed a cloud gaming platform based on graphics compression. Their platform has three graphics compression tools: (i) intra-frame compression, (ii) inter-frame compression, and (iii) caching. These tools are applied to graphics commands, 3D

structures, and 2D textures. Another work [Meilander et al. 2014] also developed a similar platform for mobile devices, where the graphics are sent from cloud servers to proxy clients, which then render game scenes for mobile devices. They also propose three graphics compression tools: (i) caching, (ii) lossy compression, and (iii) multi-layer compression. Generally speaking, tuning cloud gaming platforms based on graphics compression for heterogeneous client computers is non-trivial, because mobile (or even stationary) computers may not have enough computational power to locally render game scenes.

Hybrid compression attempts to fully utilize the available computational power on client computers to maximize the coding efficiency. Chuan et al. [2014] propose to apply graphics compression on simplified 3D structures and 2D textures, and send them to client computers. The simplified scenes are then rendered on client computers, which is called the *base* layer. Both the full-quality video and the base-layer video are rendered on cloud servers, and the residue video is compressed using video compression and sent to client computers. This is called the *enhancement* layer. Since the base layer is compressed as graphics and the enhancement layer is compressed as videos, the proposed approach is a hybrid scheme.

11.5.2 Adaptive Transmission

Even though data compression techniques have been applied to reduce the network transmission rate, the fluctuating network provisioning still results in unstable service quality to the players in a cloud gaming system. These unpredictable factors include bandwidth, round-trip time, jitter, etc. Under this circumstance, adaptive transmission is introduced to further optimize players' QoE. It is based on common sense that players would sacrifice video quality to gain smoother playing experience with poor network connections.

The first work in adaptive transmission for cloud gaming was introduced by a joint work of a Finnish research group and G-cluster in 2006. Jarvinen et al. [2006] explore the approach to adapt the gaming video transmission to available bandwidth. This is accomplished by integrating a video adaptation module into the system, which estimates the network status from a network monitor in real-time and dynamically manipulates the encoding parameters, such as frame rate and quantization, to produce a specific adaptive bit rate video stream. The authors utilize round trip time (RTT) jitter value to detect the network congestion, and thus decide if the bit rate adaptation should be triggered. To evaluate this proposal, Laulajainen et al. [2006] conducted experiments on a normal television with an Internet Protocol television (IPTV) set-top-box. The authors simulated the network scenarios in homes and hotels to verify that the proposed adaptation performed notably better.

Another series of investigations, conducted by a research group from University of California, San Diego, focus on the adaptation in mobile scenarios. Their first work decomposed the cloud gaming system's response time into sub-components: server delay, network uplink/downlink delay, and client delay [Wang and Dey 2010a]. Among the optimization techniques applied, the rate-selection algorithm best provides a dynamic solution that determines the time and the way to switch the bit rate according to the network delay. As a further step, Wang and Dey [2010b] study the potential of rendering adaptation. The authors identify the rendering parameters that affect a particular game, including realistic effect (e.g., colour depth, multi-sample, texture-filter, and lighting mode), texture detail, view distance, and enabling grass. Afterward, they analyze these parameters' characteristics of communications and computation costs and propose their rendering adaptation scheme, which consists of optimal adaptive rendering settings and level-selection algorithm. With the experiments conducted on commercial wireless networks, the authors demonstrate that acceptable mobile gaming user experience can be ensured by their rendering adaptation technique. Thus, they claim that their proposal is able to facilitate cloud gaming over the mobile network. Other works, including Wang and Dey [2009], provide more solid experiments to support their claims and further extend their application scenarios to cloud mobile rendering for rich multimedia applications.

Other aspects of transmission adaptation have also been investigated in the literature. He et al. [2014] consider adaptive transmission from the perspective of multiplayer gameplay. The authors calculate the packet urgency based on buffer status estimation and propose a scheduling algorithm. In addition, they also suggest an adaptive video segment request scheme, which estimates media access control (MAC) queue as additional information to determine the request time interval for each gamer, for the purpose of improving the playback experience.

Bujari et al. [2015] provide a Vegas over Access Point (VoAP) algorithm to address the flow coexistence issue in wireless cloud gaming service delivery. This research problem is introduced by the concurrent transmissions of TCP-based and UDP-based streams in the home scenario, where the downlink requirement of gaming video exacerbates the operation of the above-mentioned transport protocols. The authors' solution is to dynamically modify the advertised window in such a way that the system can limit the growth of the TCP flow's sending rate.

Wu et al. [2015b] presents a novel transmission scheduling framework dubbed AdaPtive HFR vIdeo Streaming (APHIS) to address the issue in cloud gaming video delivery through wireless networks. The authors first propose an online video frame selection algorithm to minimize the total distortion based on network status, input video data, and delay constraint. Afterward, they introduce an unequal forward

error correction (FEC) coding scheme to provide differentiated protection for Intra (I) and Predicted (P) frames with low latency cost. The proposed APHIS framework is able to appropriately filter video frames and adjust data protection levels to optimize the quality of high frame rate (HFR) video streaming.

Hemmati et al. [2013] proposes an object selection algorithm to provide an adaptive scene rendering solution. The basic idea is to exclude less important objects from the final output, thus consuming less processing time for the server to render and encode the frames. In such a way, the cloud gaming system is able to achieve a lower bit rate to stream the resulting video. The proposed algorithm evaluates the importance of objects from the game scene based on the analysis of gamers' activities and does the selection work. Experiments demonstrate that this approach reduces streaming bit rate by up to 8.8%.

11.6 Future Paradigm of Cloud Gaming

Cloud gaming services have gone through their initial growing pains, and it is now the key moment for cloud gaming to be deployed in living rooms everywhere. Several new cloud gaming services have been launched, including the OTT service offered by Sony PlayStation Now and the integrated service offered by Comcast XFINITY Games. With such breakthroughs, we believe that cloud gaming will undergo a series of dramatic upgrades in all aspects, and thus we present some of our forecasts on the future paradigm of cloud gaming in this section.

11.6.1 Cloud Gaming Engages More Multiplayer Games

The gaming industry has seen a shift toward games with multiplayer facilities. A larger percentage of games on all types of platforms start to incorporate some form of competition elements among online players. However, researchers in the cloud gaming field have yet to explore the huge potential of cloud gaming in multiplayer scenarios, which involve more than one player in the same game simultaneously. In such scenarios, players can interact with each other in partnership, competition, or rivalry, which provides them with opportunities for social communication that is absent in single-player games. Here we consider multiplayer games as either massively multiplayer online role-playing games (MMORPGs), e.g., World of Warcraft, Lineage, or small-scale networked games, e.g., StarCraft, Diablo, League of Legends.

Due to rapid developments in both computer and broadband network technologies, MMORPG has become an important part of the modern online entertainment industry. Game players tend to play games with a group of peers instead of cooperating or competing with the game artificial intelligence. This change enhances

the gameplay experience and makes more players attached to this type of games. Similarly, small-scale networked games also attract many players due to their flexibility to form groups and engage in short sessions of gaming. In fact, delivering multiplayer games via cloud gaming introduces additional benefits as follows:

Nature of Connectivity. A critical drawback of the cloud gaming paradigm is the indispensable network connectivity. Indeed, an overhead is incurred to establish and maintain the network connections between the cloud and players' terminals during gameplay. This limitation may keep some users away from cloud gaming. However, this concern is unlikely to impact the decisions of end users when it comes to multiplayer games, since network access is already mandatory for such games.

Temporary Engagement. An important feature of cloud gaming is to enable gameplay without download and installation. This nature of click-and-play becomes more attractive in a multiplayer scenario where people in the vicinity are engaged to play the same game with a short setup time. For instance, several friends at a party might decide to play a video game together but they cannot find a game that is installed on all of their smartphones. In this case, the benefit of click-and-play cloud gaming becomes evident.

Gaming Fairness. How to achieve fairness between multiple players is a crucial issue in the design of online games. As game players are competing with each other in real time, the server should respond to their actions immediately. Players in a conventional online game may suffer from unfairness, especially when the QoS (e.g., latency, packet loss rate) of their network connections varies. With cloud gaming, players' gaming instances are hosted in the cloud. Hence, the message exchanges between game instances occur inside the cloud, which makes it easier to maintain a guaranteed QoS level. The cloud gaming system can therefore be more capable of adapting itself to a terminal's network to provide better fairness. For example, previous research proposed to adjust rendering parameters to reduce video quality for those players with poor network access [Wang and Dey 2013]. By reducing the video quality, players with less capable devices or experiencing poor network conditions can be treated more fairly in a multiplayer game.

Research challenges related to multiplayer gaming include:

Video Sharing. Video sharing cooperative cloud gaming reduces bandwidth consumption with cooperative encoding. However, it also brings several challenges and research issues. First, reference-based encoding introduces additional overhead to the system, such as increased workload in the video

encoder servers. A common assumption in the cloud computing paradigm is that all game engines and video encoder servers must be extremely powerful to perform the encoding workload, due to scalable computing resources. Nevertheless, in practice, the cost of cloud resources cannot be neglected. Hence, optimization inside the cloud should be considered. Second, decoding video frames from predicted images requires additional cloudlet support, or using an ad hoc networking model that enables terminals to decode video frames cooperatively. Energy consumption of mobile terminals to perform the tasks of cooperative video decoding and ad hoc network communications can be a critical issue. Furthermore, system performance in the presence of device mobility can also be an important issue.

Cooperative Component Sharing. Component-sharing cooperative cloud gaming should be built upon the concept of component-based gaming architecture. The most commonly seen challenge for such an architecture is the decomposition complexity, or, to be more specific, the decomposition level (e.g., data level, task level, function level). The decomposition level defines the frequency with which components interact with each other, and thus the rate of data exchange between components. It is actually the determining factor in the ad-hoc cloudlet-based gaming architecture. Since components could be remotely executed, a high data exchange rate (high decomposition level) between remote components could be highly detrimental to both the system performance and communication cost. As the decomposition level varies with game genre, how to find the appropriate level of decomposition remains the biggest challenge. Furthermore, the beacon messages and the memory used to acquire and store the neighbors' gaming statuses are overheads that require further modeling and analysis. Moreover, efficient and decentralized service discovery, device discovery, and membership management mechanisms should be carefully designed to ensure the scalability of the system.

11.6.2 Novel Gaming Paradigm Convergence in Cloud

Cloud computing provides additional opportunities for novel game paradigms, such as virtual reality (VR) games, augmented-reality games, and context-aware games.

Virtual Reality Cloud Gaming. VR games have been talked about for years, but we have yet to see many of them become available on the market. A number of leading companies, such as Oculus, Valve, and HTC, are on their way to

producing high-quality equipment to facilitate VR games. If realized, the industry then needs to start building content for this new and potentially “game changing” platform as quickly as possible. However, the real-time rendering of omnipresent 3D scenes requires very strong graphical computation power, which might limit the application of VR games. A potential solution involves the cloud, as it provides rich resource, up in the clusters, e.g., NVIDIA GRID, etc. The high volume of memory and computational power could allow the infinite virtual world to become reality for the players.

Augmented-Reality Cloud Gaming. Project Glass is a research and development program by Google to develop an augmented-reality head-mounted display [Goldman 2012]. In contrast to traditional mobile devices, Google Glass provides a hands-free display of information on the lenses, integrating the virtual display to the reality in one’s vision. In addition, the device enables people to interact with the Internet via natural-language voice command. Therefore, it is a perfect solution for augmented-reality cloud games, which can be launched while people are walking. A typical augmented-reality cloud gaming experience can be demonstrated by the following examples. The camera on the glasses continuously captures the player’s vision in real-time, and the device transmits the video to the cloud via a wireless network. In the cloud-end, the video analyzer processes the video images with sophisticated artificial intelligence technologies, such as pattern recognition. The game logic in the cloud then creates gaming contents and delivers them to the game players. These virtual gaming contents, such as coins and bombs, can be displayed in the real scenarios through the lenses. Therefore, the system provides the players a gaming world with mixed virtual and real items. During the gaming session, the players should move their bodies or their vision angles to interact with those virtual items, in order to achieve the designed gaming goals. This type of games could also be used in daily exercise and for healthcare purposes. However, how to guarantee the safety of players during the gaming session remains a critical issue for game designers.

Context-Aware Cloud Gaming. An example of context-aware cloud gaming is gaming onboard a vehicle. People often prefer to entertain themselves with games when they are trying to pass time onboard a bus or subway train. The mobility of a vehicle provides a new gaming experience for players. In this gaming scenario, the vehicular game reports Global Positioning System (GPS) information to the cloud via a wireless network, so that the cloud is

able to deliver corresponding gaming contents to the mobile devices. For example, when the player is in the urban area, the environment of the game is set to be in the crowd and is busy; when the player is in the suburban area, virtual wild animals might appear in the game and attack the avatar. Furthermore, the game also collects the mobility information with its equipped accelerators and the cloud utilizes these sensed data to facilitate various gaming contents. For example, when the vehicle is accelerating, the avatar in the game will enter a speed-up mode, such that the player has less response time to deal with the challenges in the game. In addition, with the assistance of the cloud, the game is able to search for peer players, e.g., those in the same vehicle, thus introducing more interactive gaming scenarios such as encountered challenges.

11.7 Conclusion

In this chapter, we have briefly reviewed the history of cloud gaming services and remarked that it is a key moment for cloud gaming services to increase their penetration rates. We presented a representative open-source cloud gaming platform called GamingAnywhere, and presented research summaries in various sub-fields including cloud deployment, thin client design, and issues in communication mechanisms. Last, built upon our extensive research experience in cloud gaming, we shared several of our visions into the paradigms of cloud gaming technologies, in the format of forecasts. We hope that this chapter will serve as a useful pointer for researchers and practitioners who are new to cloud gaming and help stimulate a sustainable cloud gaming ecosystem.

Bibliography

- M. Abadi, R. Subramanian, S. Kia, P. Avesani, I. Patras, and N. Sebe. 2015. DECAF: MEG-based multimodal database for decoding affective physiological responses. *IEEE Transactions on Affective Computing*, 6(3): 209–222. DOI: [10.1109/TAFFC.2015.2392932](https://doi.org/10.1109/TAFFC.2015.2392932). 221, 238, 240, 249
- A. E. Abdel-Hakim and A. A. Farag. 2006. CSIFT: A SIFT descriptor with color invariant characteristics. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 1978–1983. DOI: [10.1109/CVPR.2006.95](https://doi.org/10.1109/CVPR.2006.95). 234
- E. Abdulin and O. Komogortsev. 2015. User eye fatigue detection via eye movement behavior. In *CHI Extended Abstracts*, pp. 1265–1270. DOI: [10.1145/2702613.2732812](https://doi.org/10.1145/2702613.2732812). 250
- E. Adam. 1993. Fighter cockpits of the future. In *Digital Avionics Systems Conference, 1993. 12th DASC., AIAA/IEEE*, pp. 318–323. IEEE. DOI: [10.1109/DASC.1993.283529](https://doi.org/10.1109/DASC.1993.283529). 165, 167
- A. Adelsbach and A.-R. Sadeghi. 2001. Zero-knowledge watermark detection and proof of ownership. In *Information Hiding*, volume 2137 of *Lecture Notes in Computer Science*, pp. 273–288. Springer. DOI: [10.1007/3-540-45496-9_20](https://doi.org/10.1007/3-540-45496-9_20). 81
- A. Adi and O. Etzion. 2004. Amit—the situation manager. *The VLDB Journal—The International Journal on Very Large Data Bases*, 13(2): 177–203. DOI: [10.1007/s00778-003-0108-y](https://doi.org/10.1007/s00778-003-0108-y). 165
- C. Aggarwal, J. H. Jiawei, and J. W. P. S. Yu. 2003. A framework for clustering evolving data streams. In *Proceedings of the VLDB Endowment 29th International Conference on Very Large Data Bases*, volume 29, pp. 81–92. 97, 98
- C. Aggarwal, D. Olshefski, D. Saha, Z.-Y. Shae, and P. Yu. 2005. CSR: Speaker recognition from compressed VoIP packet stream. In *IEEE International Conference on Multimedia and Expo*, pp. 970–973. DOI: [10.1109/ICME.2005.1521586](https://doi.org/10.1109/ICME.2005.1521586). 97, 98
- J. K. Aggarwal and M. S. Ryoo. 2011. Human activity analysis: A review. *ACM Computing Surveys*, 43(3): April 2011. DOI: [10.1145/1922649.1922653](https://doi.org/10.1145/1922649.1922653). 4
- A. Aghasaryan, M. Bouzid, D. Kostadinov, M. Kothari, and A. Nandi. July 2013. On the use of LSH for privacy preserving personalization. In *IEEE International Conference on Trust, Security, and Privacy in Computing and Communications (TrustCom)*, pp. 362–371. DOI: [10.1109/TrustCom.2013.46](https://doi.org/10.1109/TrustCom.2013.46). 105

- E. Agichtein, E. Brill, S. Dumais, and R. Ragno. 2006. Learning user interaction models for predicting web search result preferences. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3–10. ACM. DOI: [10.1145/1148170.1148175](https://doi.org/10.1145/1148170.1148175). 147
- E. Agrell, T. Eriksson, A. Vardy, and K. Zeger. 2002. Closest point search in lattices. *IEEE Trans. Inform. Theory*, 48(8): 2201–2214. DOI: [10.1109/TIT.2002.800499](https://doi.org/10.1109/TIT.2002.800499). 115
- M. Aguilar, C. XLIM, S. Fau, C. Fontaine, G. Gogniat, and R. Sirdey. 2013. Recent advances in homomorphic encryption: A possible future for signal processing in the encrypted domain. *IEEE Signal Processing Magazine*, 30(2): 108–117. DOI: [10.1109/MSP.2012.2230219](https://doi.org/10.1109/MSP.2012.2230219). 77
- N. Ahituv, Y. Lapid, and S. Neumann. 1987. Processing encrypted data. *Communications of the ACM*, 30(9): 777–780. DOI: [10.1145/30401.30404](https://doi.org/10.1145/30401.30404). 78, 84
- L. Ai, J. Yu, Z. Wu, Y. He, and T. Guan. 2015. Optimized residual vector quantization for efficient approximate nearest neighbor search. *Multimedia Systems*, pp. 1–13. DOI: [10.1007/s00530-015-0470-9](https://doi.org/10.1007/s00530-015-0470-9). 131
- X. Alameda-Pineda and R. Horaud. 2015. Vision-guided robot hearing. *International Journal of Robotics Research*, 34(4–5): 437–456. DOI: [10.1177/0278364914548050](https://doi.org/10.1177/0278364914548050). 55
- X. Alameda-Pineda, J. Sanchez-Riera, J. Wienke, V. Franc, J. Cech, K. Kulkarni, A. Deleforge, and R. Horaud. 2013. Ravel: An annotated corpus for training robots with audiovisual abilities. *Journal on Multimodal User Interfaces*, 7(1–2): 79–91. DOI: [10.1007/s12193-012-0111-y](https://doi.org/10.1007/s12193-012-0111-y). 55
- X. Alameda-Pineda, Y. Yan, E. Ricci, O. Lanz, and N. Sebe. 2015. Analyzing free-standing conversational groups: A multimodal approach. In *ACM International Conference on Multimedia (ACMMM)*, pp. 4–15. DOI: [10.1145/2733373.2806238](https://doi.org/10.1145/2733373.2806238). 55, 56, 61
- X. Alameda-Pineda, E. Ricci, Y. Yan, and N. Sebe. 2016a. Recognizing emotions from abstract paintings using non-linear matrix completion. In *CVPR*. 58
- X. Alameda-Pineda, J. Staiano, R. Subramanian, L. Batrinca, E. Ricci, B. Lepri, O. Lanz, and N. Sebe. 2016b. SALSAs: A novel dataset for multimodal group behavior analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence, (IEEE TPAMI)*, 38(8): 1707–1720. DOI: [10.1109/TPAMI.2015.2496269](https://doi.org/10.1109/TPAMI.2015.2496269). 56, 66, 238
- M. Alhamad, T. Dillon, and E. Chang. 2010. SLA-based trust model for cloud computing. In *Proceedings of IEEE International Conference on Network-Based Information Systems (NBIS)*, pp. 321–324. DOI: [10.1109/NBiS.2010.67](https://doi.org/10.1109/NBiS.2010.67). 260
- Y. Aloimonos, A. K. Mishra, L. F. Cheong, and A. Kassim. 2012. Active visual segmentation. *IEEE PAMI*, 34: 639–653. DOI: [10.1109/TPAMI.2011.171](https://doi.org/10.1109/TPAMI.2011.171). 220
- D. Anderson. 2004. BOINC: A system for public-resource computing and storage. In *Proceedings of IEEE/ACM Grid Computing (GC)*, pp. 4–10. DOI: [10.1109/GRID.2004.14](https://doi.org/10.1109/GRID.2004.14). 258, 261

- D. Anderson, J. Cobb, E. Korpela, M. Lebofsky, and D. Werthimer. 2002. SETI@home: An experiment in public-resource computing. *ACM Transactions on Communications*, 45(11): 56–61. DOI: [10.1145/581571.581573](https://doi.org/10.1145/581571.581573). 258
- N. Anderson, W. F. Bischof, K. E. Laidlaw, E. F. Risko, and A. Kingstone. 2013. Recurrence quantification analysis of eye movements. *Behavior Research Methods*, 45(3): 842–856. DOI: [10.3758/s13428-012-0299-5](https://doi.org/10.3758/s13428-012-0299-5). 229
- A. Andoni. Nov. 2009. *Nearest Neighbor Search: The Old, the New, and the Impossible*. PhD thesis, MIT. 108, 113, 115
- A. Andoni and P. Indyk. 2006. Near-optimal hashing algorithms for near neighbor problem in high dimensions. In *Proceedings of the Symposium on the Foundations of Computer Science*, pp. 459–468. DOI: [10.1109/FOCS.2006.49](https://doi.org/10.1109/FOCS.2006.49). 105, 112, 115, 117
- P. André, E. Cutrell, D. S. Tan, and G. Smith. 2009. Designing novel image search interfaces by understanding unique characteristics and usage. In *IFIP Conference on Human-Computer Interaction*, pp. 340–353. Springer. 148
- M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2014. 2D human pose estimation: New benchmark and state of the art analysis. In *CVPR*, pp. 3686–3693. DOI: [10.1109/CVPR.2014.471](https://doi.org/10.1109/CVPR.2014.471). 22
- S. Argamon, S. Dhawle, M. Koppel, and J. Pennbaker. 2005. Lexical predictors of personality type. In *Interface and the Classification Society of North America*. DOI: [10.1.1.60.6697.238](https://doi.org/10.1.1.60.6697.238)
- V. Athitsos, J. Alon, S. Sclaroff, and G. Kollios. Jan. 2008. BoostMap: An embedding method for efficient nearest neighbor retrieval. *IEEE Trans. PAMI*, 30(1): 89–104. DOI: [10.1109/TPAMI.2007.1140](https://doi.org/10.1109/TPAMI.2007.1140). 133
- S. Avidan and M. Butman. 2006. Blind vision. In *Proceedings of the 9th European Conference on Computer Vision*, volume 3953 of *Lecture Notes in Computer Science*, pp. 1–13. Springer. DOI: [10.1007/11744078_1](https://doi.org/10.1007/11744078_1). 82, 91, 93
- G. Awad, J. Fiscus, M. Michel, D. Joy, W. Kraaij, A. F. Smeaton, G. Quenot, M. Eskevich, R. Aly, G. J. F. Jones, R. Ordelman, B. Huet, and M. Larson. 2016. TRECVID 2016: Evaluating video search, video event detection, localization, and hyperlinking. In *Proceedings of TRECVID 2016*, pp. 407–411. 23, 27
- S. Ba, X. Alameda-Pineda, A. Xompero, and R. Horaud. 2016. An on-line variational Bayesian model for multi-person tracking from cluttered scenes. In *Computer Vision and Human Understanding (CVHU)* 153:64–76. DOI: [10.1016/j.cviu.2015.07.006](https://doi.org/10.1016/j.cviu.2015.07.006). 52, 55
- O. Babaoglu, M. Marzolla, and M. Tamburini. 2012. Design and implementation of a P2P cloud system. In *Proceedings of ACM Symposium on Applied Computing (SAC)*, pp. 412–417. DOI: [10.1145/2245276.2245357](https://doi.org/10.1145/2245276.2245357). 259
- A. Babenko and V. Lempitsky. June 2014. Additive quantization for extreme vector compression. In *CVPR*. DOI: [10.1109/CVPR.2014.124](https://doi.org/10.1109/CVPR.2014.124). 131
- A. Babenko and V. Lempitsky. June 2015a. The inverted multi-index. In *IEEE TPAMI*, 37(6): 1247–1260. DOI: [10.1109/CVPR.2012.6248038](https://doi.org/10.1109/CVPR.2012.6248038). 132

- A. Babenko and V. Lempitsky. June 2015b. Tree quantization for large-scale similarity search and classification. In *CVPR*. DOI: [10.1109/CVPR.2015.7299052](https://doi.org/10.1109/CVPR.2015.7299052). 131
- A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. September 2014. Neural codes for image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 584–599. DOI: [10.1007/978-3-319-10590-1_38](https://doi.org/10.1007/978-3-319-10590-1_38). 107
- M. Backes, G. Doychev, M. Dürmuth, and B. Köpf. 2010. Speaker recognition in encrypted voice streams. In *Computer Security—ESORICS*, volume 6345 of *Lecture Notes in Computer Science*, pp. 508–523. Springer. 98, 100
- A. Baddeley and R. Turner. 2000. Practical maximum pseudolikelihood for spatial point patterns. *Australian & New Zealand Journal of Statistics*, 42: 283–322. DOI: [10.1111/1467-842X.00128](https://doi.org/10.1111/1467-842X.00128). 208
- S. Bahmani and B. Raj. 2013. A unifying analysis of projected gradient descent for ℓ_p -constrained least squares. *Applied and Computational Harmonic Analysis*, 34(3): 366–378. DOI: [10.1016/j.acha.2012.07.004](https://doi.org/10.1016/j.acha.2012.07.004). 39
- S. Bahmani, P. Boufounos, and B. Raj. 2011. Greedy sparsity-constrained optimization. In *Proceedings 45th IEEE Asilomar Conference on Signals, Systems, and Computers (ASILOMAR)*, pp. 1148–1152. 39
- S. Bahmani, B. Raj, and P. Boufounos. 2016. Learning model-based sparsity via projected gradient descent. *IEEE Transactions on Information Theory*, 62(4): 2092–2099. DOI: [10.1109/TIT.2016.2515078](https://doi.org/10.1109/TIT.2016.2515078). 39
- N. Ballas, L. Yao, C. Pal, and A. Courville. 2016. Delving deeper into convolutional networks for learning video representations. In *Proceedings of the International Conference on Learning Representations (ICLR)*. *arXiv:1511.06432, 2015*. 14, 28
- T. Baltrušaitis, P. Robinson, and L.-P. Morency. 2016. OpenFace: an open source facial behavior analysis toolkit. In *IEEE Winter Conference on Applications of Computer Vision*, pp. 1–10. DOI: [10.1109/WACV.2016.7477553](https://doi.org/10.1109/WACV.2016.7477553). 225
- R. Balu, T. Furon, and H. Jégou. April 2014. Beyond project and sign for distance estimation with binary codes. In *Proc IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6884–6888. DOI: [10.1109/ICASSP.2014.6854934](https://doi.org/10.1109/ICASSP.2014.6854934). 123, 127
- S. Banerjee and A. Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72. 27
- K. Baokye, B. Trueba-Hornero, O. Vinyals, and G. Friedland. 2008a. Overlapped speech detection for improved diarization in multiparty meetings. In *ICASSP*, pp. 4353–4356. DOI: [10.1109/ICASSP.2008.4518619](https://doi.org/10.1109/ICASSP.2008.4518619). 36
- K. Baokye, O. Vinyals, and G. Friedland. 2008b. Two’s a crowd: Improving speaker diarization by automatically identifying and excluding overlapped speech. In *Proceedings of Interspeech*, pp. 32–35. 36

- R. Bardeli, D. Wolff, F. Kurth, M. Koch, K.-H. Tauchert, and K.-H. Frommolt. 2010. Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring. *Pattern Recognition Letters*, 31(12): 1524–1534. DOI: [10.1016/j.patrec.2009.09.014](https://doi.org/10.1016/j.patrec.2009.09.014). 38
- M. Barni and A. Piva. 2008. Co-chairs, special issue on signal processing in the encrypted domain. *Proceedings of EURASIP European Signal Processing Conference*. 77
- M. Barni, T. Bianchi, D. Catalano, M. Di Raimondo, R. D. Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, A. Piva, and F. Scotti. 2010. A privacy-compliant fingerprint recognition system based on homomorphic encryption and fingerprint templates. In *4th IEEE International Conference on Biometrics: Theory Applications and Systems*, pp. 1–7. DOI: [10.1109/BTAS.2010.5634527](https://doi.org/10.1109/BTAS.2010.5634527). 82
- M. Barni, P. Failla, R. Lazzeretti, A. R. Sadeghi, and T. Schneider. 2011. Privacy-preserving ECG classification with branching programs and neural networks. *IEEE Transactions on Information Forensics and Security*, 6(2): 452–468. DOI: [10.1109/TIFS.2011.2108650](https://doi.org/10.1109/TIFS.2011.2108650). 83
- R. Barthes. 1981. *Camera Lucida: Reflections on Photography*. Macmillan. 187
- J. Barwise and J. Perry. 1980. The situation underground. *Stanford Working Papers in Semantics*, 1: 1–55. 165, 167
- J. Barwise and J. Perry. 1981. Situations and attitudes. *The Journal of Philosophy*, 78(11): 668–691. DOI: [10.2307/2026573](https://doi.org/10.2307/2026573). 165
- R. Bayer and E. M. McCreight. July 1970. Organization and maintenance of large ordered indices. Mathematical and Information Sciences Report 20, Boeing Scientific Research Laboratories. DOI: [10.1145/1734663.1734671](https://doi.org/10.1145/1734663.1734671). 107
- M. Bebbington and D. S. Harte. 2001. On the statistics of the linked stress release model. *Journal of Applied Probability*, 38(A): 176–187. DOI: [10.1017/S0021900200112768](https://doi.org/10.1017/S0021900200112768). 208
- J. S. Beis and D. G. Lowe. June 1997. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *CVPR*. DOI: [10.1109/CVPR.1997.609451](https://doi.org/10.1109/CVPR.1997.609451). 105, 107, 118
- A. J. Bell and T. J. Sejnowski. 1995. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6): 1129–1159. citeseer.nj.nec.com/bell95informationmaximization.html. 36
- A. Beloglazov, J. Abawajy, and R. Buyya. 2012. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, 28(5): 755–768. DOI: [10.1016/j.future.2011.04.017](https://doi.org/10.1016/j.future.2011.04.017). 259
- J. Benaloh. 1994. Dense probabilistic encryption. In *Proceedings of the Workshop on Selected Areas of Cryptography*, pp. 120–128. 81
- B. Benfold and I. Reid. 2011. Unsupervised learning of a scene-specific coarse gaze estimator. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2344–2351. DOI: [10.1109/ICCV.2011.6126516](https://doi.org/10.1109/ICCV.2011.6126516). 57

- Y. Bengio, P. Simard, and P. Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks (TNN)*, 5(2): 157–166. DOI: [10.1109/72.279181](https://doi.org/10.1109/72.279181). 7
- Y. Bengio, A. Courville, and P. Vincent. 2013. Representation learning: A review and new perspectives. *IEEE TPAMI*, pp. 1798–1828. DOI: [10.1109/TPAMI.2013.50](https://doi.org/10.1109/TPAMI.2013.50). 6
- C. J. Bennett. 2008. *The Privacy Advocates: Resisting the Spread of Surveillance*. MIT Press. 91
- W. Bennett. July 1948. Spectra of quantized signals. *Bell System Technical Journal*, 27(3): 1–43. DOI: [10.1002/j.1538-7305.1948.tb01340.x](https://doi.org/10.1002/j.1538-7305.1948.tb01340.x). 117
- A. Berenzweig, B. Logan, D. P. W. Ellis, and B. Whitman. June 2004. A large-scale evaluation of acoustic and subjective music-similarity measures. *Computer Music Journal*, 28(2): 63–76. <http://www.ee.columbia.edu/~dpwe/pubs/BerenLEW04-museval.pdf>. DOI: [10.1162/014892604323112257](https://doi.org/10.1162/014892604323112257). 38
- T. Bertin-Mahieux and D. P. W. Ellis. Oct. 2011. Large-scale cover song recognition using hashed chroma landmarks. In *Proceedings IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 117–120. DOI: [10.7916/D8J67S98](https://doi.org/10.7916/D8J67S98). 38
- T. Bianchi, A. Piva, and M. Barni. 2008a. Efficient pointwise and blockwise encrypted operations. In *Proceedings of the ACM 10th Workshop on Multimedia and Security*, pp. 85–90. DOI: [10.1145/1411328.1411344](https://doi.org/10.1145/1411328.1411344). 79, 80
- T. Bianchi, A. Piva, and M. Barni. 2008b. Comparison of different FFT implementations in the encrypted domain. In *Proceedings of EUSIPCO 16th European Signal Processing Conference*, pp. 1–5. 80
- T. Bianchi, A. Piva, and M. Barni. 2008c. Implementing the discrete Fourier transform in the encrypted domain. In *IEEE International Conference on Acoustics*, pp. 1757–1760. 80
- T. Bianchi, A. Piva, and M. Barni. 2009a. Efficient linear filtering of encrypted signals via composite representation. In *Proceedings of the IEEE 16th International Conference on Digital Signal Processing*, pp. 1–6. DOI: [10.1109/ICDSP.2009.5201116](https://doi.org/10.1109/ICDSP.2009.5201116). 80
- T. Bianchi, T. Veugen, A. Piva, and M. Barni. 2009b. Processing in the encrypted domain using a composite signal representation: PROS and CONS. In *Proceedings of the IEEE 1st International Workshop on Information Forensics and Security*, pp. 176–180. DOI: [10.1109/WIFS.2009.5386460](https://doi.org/10.1109/WIFS.2009.5386460). 76, 77, 79, 80
- T. Bianchi, S. Turchi, A. Piva, R. D. Labati, V. Piuri, and F. Scotti. 2010. Implementing fingerprint-based identity matching in the encrypted domain. In *IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications*, pp. 15–21. DOI: [10.1109/BIOMS.2010.5610445](https://doi.org/10.1109/BIOMS.2010.5610445). 77, 82
- H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould. 2016. Dynamic image networks for action recognition. In *CVPR*, pp. 3034–3042. 12
- D. Bitouk, N. Kumar, S. Dhillon, P. Belhumeur, and S. Nayar. 2008. Face swapping: automatically replacing faces in photographs. In *ACM Transactions on Graphics*, volume 27, p. 39. DOI: [10.1145/1399504.1360638](https://doi.org/10.1145/1399504.1360638). 92, 93

- M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. 2005. Actions as space-time shapes. In *ICCV*. DOI: [10.1109/TPAMI.2007.70711](https://doi.org/10.1109/TPAMI.2007.70711). 20
- D. Bogdanov, 2007. Foundations and properties of Shamir's secret sharing scheme. *Research Seminar in Cryptography*. 88
- F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. 2012. Fog computing and its role in the Internet of Things. In *Proceedings of ACM Workshop on Mobile Cloud Computing (MCC)*, pp. 13–16. DOI: [10.1145/2342509.2342513](https://doi.org/10.1145/2342509.2342513). 256
- D. Borth, R. Ji, T. Chen, T. Breuel, and S.-F. Chang. 2013. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 223–232. ACM. DOI: [10.1145/2502081.2502282](https://doi.org/10.1145/2502081.2502282). 150
- P. T. Boufounos. March 2012. Universal rate-efficient scalar quantization. *IEEE Trans. Inform. Theory*, 58(3): 1861–1872. DOI: [10.1109/TIT.2011.2173899](https://doi.org/10.1109/TIT.2011.2173899). 122
- P. T. Boufounos and R. G. Baraniuk. March 2008. 1-bit compressive sensing. In *Proceedings Conference on Information Science and Systems (CISS)*, pp. 16–21. DOI: [10.1109/CISS.2008.4558487](https://doi.org/10.1109/CISS.2008.4558487). 121
- T. E. Boul. 2005. Pico: Privacy through invertible cryptographic obscuration. In *Proceedings of IEEE Computer Vision for Interactive and Intelligent Environment*, pp. 27–38. DOI: [10.1109/CVIE.2005.16](https://doi.org/10.1109/CVIE.2005.16). 91
- A. Bourrier, F. Perronnin, R. Gribonval, P. Pérez, and H. Jégou. 2015. Nearest neighbor search for arbitrary kernels with explicit embeddings. *Journal of Mathematical Imaging and Vision*, 52(3): 459–468. 133
- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1): 1–122. DOI: [10.1561/2200000016](https://doi.org/10.1561/2200000016). 63, 64
- G. Bradski and A. Kaehler. 2008. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media. 179
- Z. Brakerski, C. Gentry, and V. Vaikuntanathan. 2012. (Leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of 3rd ACM Innovations in Theoretical Computer Science Conference*, pp. 309–325. DOI: [10.1145/2090236.2090262](https://doi.org/10.1145/2090236.2090262). 101, 102
- J. Brandt. June 2010. Transform coding for fast approximate nearest neighbor search in high dimensions. In *CVPR*, pp. 1815–1822. DOI: [10.1109/CVPR.2010.5539852](https://doi.org/10.1109/CVPR.2010.5539852). 128
- O. Brdiczka, P. Yuen, S. Zaidenberg, P. Reignier, and J. Crowley. 2006. Automatic acquisition of context models and its application to video surveillance. In *18th International Conference on Pattern Recognition, 2006 (ICPR)*, volume 1, pp. 1175–1178. IEEE. DOI: [10.1109/ICPR.2006.292](https://doi.org/10.1109/ICPR.2006.292). 165
- P. Brémaud and L. Massoulié. 1996. Stability of nonlinear Hawkes processes. *The Annals of Probability*, 24(3): 1563–1588. DOI: [10.1214/aop/1065725193](https://doi.org/10.1214/aop/1065725193). 200

- J. Bringer, H. Chabanne, M. Favre, A. Patey, T. Schneider, and M. Zohner. 2014. GSHADE: Faster privacy-preserving distance computation and biometric identification. In *Proceedings of the 2nd ACM Workshop on Information Hiding and Multimedia Security*, pp. 187–198. ACM. DOI: [10.1145/2600918.2600922](https://doi.org/10.1145/2600918.2600922). 82
- A. Z. Broder. June 1997. On the resemblance and containment of documents. In *Proceedings Compression and Complexity of Sequences*, p. 21. DOI: [10.1109/SEQUEN.1997.666900](https://doi.org/10.1109/SEQUEN.1997.666900). 120
- G. J. Brown and M. P. Cooke. 1994. Computational auditory scene analysis. *Computer Speech and Language*, 8: 297–336. DOI: [10.1006/csla.1994.1016](https://doi.org/10.1006/csla.1994.1016). 36, 37
- A. Bujari, M. Massaro, and C. E. Palazzi. Dec. 2015. Vegas over access point: Making room for thin client game systems in a wireless home. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(12): 2002–2012. DOI: [10.1109/TCSVT.2015.2450332](https://doi.org/10.1109/TCSVT.2015.2450332). 309
- R. Buyya, S. Garg, and R. Calheiros. 2011. SLA-oriented resource provisioning for cloud computing: Challenges. In *Proceedings of IEEE International Conference on Cloud and Service Computing (CSC)*, pp. 1–10. DOI: [10.1109/CSC.2011.6138522](https://doi.org/10.1109/CSC.2011.6138522). 259
- R. Cabral, F. De la Torre, J. Costeira, and A. Bernardino. 2014. Matrix completion for multi-label image classification. *IEEE TPAMI*, 37(1): 121–135. DOI: [10.1109/TPAMI.2014.2343234](https://doi.org/10.1109/TPAMI.2014.2343234). 58, 60
- W. Cai, M. Chen, and V. C. M. Leung. May 2014. Toward gaming as a service. *IEEE Internet Computing*, 18(3): 12–18. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6818918>. DOI: [10.1109/MIC.2014.22](https://doi.org/10.1109/MIC.2014.22). 290, 291
- W. Cai, R. Shea, C.-Y. Huang, K.-T. Chen, J. Liu, V. C. M. Leung, and C.-H. Hsu. April 2016a. The future of cloud gaming. *Proceedings of IEEE*, 104(4): 687–691. DOI: [10.1109/JPROC.2016.2539418](https://doi.org/10.1109/JPROC.2016.2539418). 304
- W. Cai, R. Shea, C.-Y. Huang, K.-T. Chen, J. Liu, V. C. M. Leung, and C.-H. Hsu. Jan. 2016b. A survey on cloud gaming: Future of computer games. *IEEE Access*, pp. 1–25. DOI: [10.1109/ACCESS.2016.2590500](https://doi.org/10.1109/ACCESS.2016.2590500). 289, 291
- A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson. 2006. People-centric urban sensing. In *Int. Work. on Wireless Internet*, article 18. DOI: [10.1145/1234161.1234179](https://doi.org/10.1145/1234161.1234179). 55
- E. J. Candès and T. Tao. 2010. The power of convex relaxation: Near-optimal matrix completion. *IEEE Trans. Inf. Theory*, 56(5): 2053–2080. DOI: [10.1109/TIT.2010.2044061](https://doi.org/10.1109/TIT.2010.2044061). 61
- J. Canny. 2002. Collaborative filtering with privacy. In *Proceedings of IEEE Symposium on Security and Privacy*, pp. 45–57. DOI: [10.1145/564376.564419](https://doi.org/10.1145/564376.564419). 82
- C. Canton-Ferrer, C. Segura, J. R. Casas, M. Pardo, and J. Hernando. 2008. Audiovisual head orientation estimation with particle filtering in multisensor scenarios. *EURASIP Journal on Advances in Signal Processing*, article 32. 57

- J. Cao, Y.-D. Zhang, Y.-C. Song, Z.-N. Chen, X. Zhang, and J.-T. Li. 2009. MCG-WEBV: A benchmark dataset for web video analysis. *Technical Report, CAS Institute of Computing Technology*, pp. 1–10. DOI: [10.1155/2008/276846](https://doi.org/10.1155/2008/276846). 21
- N. Carrier, T. Deutsch, C. Gruber, M. Heid, and L. Jarrett. Aug. 2008. The business case for enterprise mashups. IBM White Paper. 171
- M. Casey and M. Slaney. April 2007. Fast recognition of remixed music audio. In *ICASSP*, volume 4, pp. 1425–1428. DOI: [10.1109/ICASSP.2007.367347](https://doi.org/10.1109/ICASSP.2007.367347). 105
- C. Castelluccia, A. Chan, E. Mykletun, and G. Tsudik. 2009. Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM Transactions on Sensor Networks*, 5(3): 20. 103
- S. Chachada and C.-C. J. Kuo. 2014. Environmental sound recognition: A survey. *APSIPA Transactions on Signal and Information Processing*, 3. ISSN 2048-7703. http://journals.cambridge.org/article_S2048770314000122. DOI: [10.1017/ATSIP.2014.12](https://doi.org/10.1017/ATSIP.2014.12). 38
- S.-F. Chang. 2013. How far we've come: Impact of 20 years of multimedia information retrieval. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 9(1s): 42. DOI: [10.1109/CCGRID.2010.132](https://doi.org/10.1109/CCGRID.2010.132). 151, 155
- V. Chang, D. Bacigalupo, G. Wills, and D. Roure. 2010. Categorisation of cloud computing business models. In *Proceedings of IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGRID)*, pp. 509–512. DOI: [10.1145/2491844](https://doi.org/10.1145/2491844). 259
- Y.-C. Chang, K.-T. Chen, C.-C. Wu, and C.-L. Lei. 2008. Inferring speech activity from encrypted Skype traffic. In *Proceedings of IEEE Global Telecommunications Conference*, pp. 1–5. 97, 98
- Y.-C. Chang, P.-H. Tseng, K.-T. Chen, and C.-L. Lei. May 2011. Understanding the performance of thin-client gaming. In *Proceedings of IEEE Communications Quality and Reliability (CQR) 2011*, pp. 1–6. DOI: [10.1109/CQR.2011.5996092](https://doi.org/10.1109/CQR.2011.5996092). 291
- M. S. Charikar. May 2002. Similarity estimation techniques from rounding algorithms. In *(Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC))*, pp. 380–388. DOI: [10.1145/509907.509965](https://doi.org/10.1145/509907.509965). 105, 110, 112, 120, 121, 123, 133
- A. Chattopadhyay and T. E. Boult. 2007. PrivacyCam: A privacy preserving camera using uCLinux on the Blackfin DSP. In *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. DOI: [10.1109/CVPR.2007.383413](https://doi.org/10.1109/CVPR.2007.383413). 91, 93
- S. Chaudhuri. 2013. *Structured Models for Semantic Analysis of Audio Content*. PhD thesis, Carnegie Mellon University. 46, 47
- S. Chaudhuri and B. Raj. 2011. Learning contextual relevance of audio segments using discriminative models over AUD sequences. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 197–200. DOI: [10.1109/ASPAA.2011.6082335](https://doi.org/10.1109/ASPAA.2011.6082335). 39
- S. Chaudhuri and B. Raj. 2012. Unsupervised structure discovery for semantic analysis of audio. In *Proceedings of Neural Information Processing Systems (NIPS)*, 2:1178–1186. 39, 46, 47

- S. Chaudhuri, M. Harvilla, and B. Raj. 2011. Unsupervised learning of acoustic unit descriptors for audio content representation and classification. In *Proceedings of Interspeech*, pp. 2265–2268. 39
- S. Chaudhuri, R. Singh, and B. Raj. 2012. Exploiting temporal sequence structure for semantic analysis of multimedia. In *Proceedings of Interspeech*, 2:1726–1729. 39, 46, 47
- C. Chen and J.-M. Odobez. 2012. We are not contortionists: Coupled adaptive learning for head and body orientation estimation in surveillance video. In *CVPR*, pp. 1544–1551. DOI: [10.1109/CVPR.2012.6247845](https://doi.org/10.1109/CVPR.2012.6247845). 53, 57, 58, 62, 68, 70, 71
- D. L. Chen and W. B. Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 190–200. 24, 27
- K.-T. Chen, Y.-C. Chang, P.-H. Tseng, C.-Y. Huang, and C.-L. Lei. Nov. 2011. Measuring the latency of cloud gaming systems. In *Proceedings of ACM Multimedia 2011*, pp. 1269–1272. DOI: [10.1145/2072298.2071991](https://doi.org/10.1145/2072298.2071991). 291, 292
- K.-T. Chen, C.-Y. Huang, and C.-H. Hsu. 2014. Cloud gaming onward: Research opportunities and outlook. *Proceedings of the 2014 IEEE International Conference on Multimedia and Expo (ICME2014)*, pp. 1–4. DOI: [10.1109/ICMEW.2014.6890683](https://doi.org/10.1109/ICMEW.2014.6890683). 290, 291
- X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv:1504.00325*. 27
- Y. Chen, T. Guan, and C. Wang. 2010. Approximate nearest neighbor search by residual vector quantization. *Sensors*, 10(12): 11259–11273. DOI: [10.3390/s101211259](https://doi.org/10.3390/s101211259). 131
- G. Chéron, I. Laptev, and C. Schmid. 2015. P-CNN: Pose-based CNN features for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3218–3226. *arXiv:1506.03607 [cs.CV]* 53
- Y. W. Ching and P. C. Su. 2009. A region of interest rate-control scheme for encoding traffic surveillance videos. *Proceedings of the 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 194–197. DOI: [10.1109/IIH-MSP.2009.114](https://doi.org/10.1109/IIH-MSP.2009.114). 92
- J. Choi, H. Lei, and G. Friedland. September 2011. The 2011 ICSI video location estimation system. *Proceedings of MediaEval 2011*. <http://www.icsi.berkeley.edu/pubs/speech/icsivideolocation11.pdf> 39
- T. Choudhury and A. Pentland. 2003. Sensing and modeling human networks using the sociometer. In *Inter. Sym. on Wearable Comp.*, p. 216. DOI: [10.1109/ISWC.2003.1241414](https://doi.org/10.1109/ISWC.2003.1241414). 53
- K.-Y. Chu, Y.-H. Kuo, and W. H. Hsu. 2013. Real-time privacy-preserving moving object detection in the cloud. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 597–600. 94, 96

- W.-T. Chu and F.-C. Chang. 2015. A privacy-preserving bipartite graph matching framework for multimedia analysis and retrieval. In *Proceedings of the 5th ACM International Conference on Multimedia Retrieval*, pp. 243–250. ACM. DOI: [10.1145/2671188.2749286](https://doi.org/10.1145/2671188.2749286). 86, 90
- S.-P. Chuah, C. Yuen, and N.-M. Cheung. 2014. Cloud gaming: A green solution to massive multiplayer online games. *IEEE Wireless Communications* (August): 78–87. DOI: [10.1109/MWC.2014.6882299](https://doi.org/10.1109/MWC.2014.6882299). 290, 291, 308
- M. Claypool and K. Claypool. 2006. Latency and player actions in online games. *ACM Communications* 49(11): 40–45. DOI: [10.1145/1167838.1167860](https://doi.org/10.1145/1167838.1167860). 256, 302
- J. Conway and N. Sloane. 1982a. Fast quantizing and decoding algorithms for lattice quantizers and codes. *IEEE Transactions on Information Theory*, 28(2): 227–232. DOI: [10.1109/TIT.1982.1056484](https://doi.org/10.1109/TIT.1982.1056484). 115
- J. Conway and N. Sloane. 1982b. Voronoi regions of lattices, second moments of polytopes, and quantization. *IEEE Transactions on Information Theory*, 28(2): 211–226. DOI: [10.1109/TIT.1982.1056483](https://doi.org/10.1109/TIT.1982.1056483). 115
- J. Conway and N. Sloane. 1990. *Sphere Packings, Lattices and Groups*, third ed. Springer. ISBN:0-387-96617-X 128
- M. Cooke, J. Barker, S. Cunningham, and X. Shao. 2006. An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120: 2421. DOI: [10.1121/1.2229005](https://doi.org/10.1121/1.2229005). 37
- P. T. Costa and R. R. McCrae. 1980. Influence of extraversion and neuroticism on subjective well-being: Happy and unhappy people. *Journal of Personality and Social Psychology*, 38(4): 668. DOI: [10.1037/0022-3514.38.4.668](https://doi.org/10.1037/0022-3514.38.4.668). 244
- P. T. J. Costa and R. R. McCrae. 1992. *NEO-PI-R Professional Manual: Revised NEO Personality and NEO Five-Factor Inventory (NEO-FFI)*, volume 4. Psychological Assessment Resources, Odessa, Florida. 237
- I. Cox, J. Kilian, T. Leighton, and T. Shamoon. 1996. A secure, robust watermark for multimedia. In *Information Hiding*, volume 1174 of *Lecture Notes in Computer Science*, pp. 185–206. Springer. ISBN:3-540-61996-8. 81
- R. Cramer, R. Gennaro, and B. Schoenmakers. 1997. A secure and optimally efficient multi-authority election scheme. *Wiley Online Library, European Transactions on Telecommunications*, 8(5): 481–490. 81
- M. Cristani, L. Bazzani, G. Paggetti, A. Fossati, A. Del Blue, G. Menegaz, and V. Murino. 2011. Social interaction discovery by statistical analysis of F-formations. In *British Machine Vision Conference (BMVC)*, 2(4). DOI: [10.5244/C.25.23](https://doi.org/10.5244/C.25.23). 52, 57, 69, 70, 71
- P. Cui, F. Wang, S. Liu, M. Ou, S. Yang, and L. Sun. 2011. Who should share what?: Item-level social influence prediction for users and posts ranking. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 185–194. ACM. DOI: [10.1145/2009916.2009945](https://doi.org/10.1145/2009916.2009945). 148

- P. Cui, S.-W. Liu, W.-W. Zhu, H.-B. Luan, T.-S. Chua, and S.-Q. Yang. 2014a. Social-sensed image search. *ACM Transactions on Information Systems (TOIS)*, 32(2): 8. DOI: [10.1145/2590974](https://doi.org/10.1145/2590974). 148
- P. Cui, Z. Wang, and Z. Su. 2014b. What videos are similar with you?: Learning a common attributed representation for video recommendation. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 597–606. ACM. DOI: [10.1145/2647868.2654946](https://doi.org/10.1145/2647868.2654946). 140
- N. Dalal and B. Triggs. 2005. Histograms of oriented gradients for human detection. In *CVPR*, pp. 886–893. 59
- D. J. Daley and D. Vere-Jones. 2003. *An Introduction to the Theory of Point Processes*, 2nd ed. Springer-Verlag, New York. DOI: [10.1007/b97277](https://doi.org/10.1007/b97277). 193, 200, 206, 208, 209
- I. Damgard and M. Jurik. 2001. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In *Proceedings of 4th International Workshop on Practice and Theory in Public Key Cryptosystems*, volume 1992 of *Lecture Notes in Computer Science*, pp. 119–136. Springer. 81, 96
- Q. Danfeng, S. Gammeter, L. Bossard, T. Quack, and L. V. Gool. 2011. Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In *CVPR*, pp. 777–784. DOI: [10.1109/CVPR.2011.5995373](https://doi.org/10.1109/CVPR.2011.5995373). 106
- P. Das, C. Xu, R. F. Doell, and J. J. Corso. 2013. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, pp. 2634–2641. DOI: [10.1109/CVPR.2013.340](https://doi.org/10.1109/CVPR.2013.340). 26
- A. Dassios and H. Zhao. 2011. A dynamic contagion process. *Advances in Applied Probability*, pp. 814–846. DOI: [10.1239/aap/1316792671](https://doi.org/10.1239/aap/1316792671). 200
- A. Dassios and H. Zhao. 2013. Exact simulation of Hawkes process with exponentially decaying intensity. *Electronic Communications in Probability*, 18: 1–13. DOI: [10.1214/ECP.v18-2717](https://doi.org/10.1214/ECP.v18-2717). 202, 204
- M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni. 2004. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the Symposium on Computational Geometry*, pp. 253–262. DOI: [10.1145/997817.997857](https://doi.org/10.1145/997817.997857). 105, 108, 113, 114, 131
- R. Datta, D. Joshi, J. Li, and J. Z. Wang. 2008. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2): 5. DOI: [10.1145/1348246.1348248](https://doi.org/10.1145/1348246.1348248). 84, 86
- M. de Berg, M. van Kreveld, M. H. Overmars, and O. Schwarzkopf. March 2008. *Computational Geometry: Algorithms and Applications*, 3rd ed. Springer-Verlag. 109
- M. De Choudhury, Y.-R. Lin, H. Sundaram, K. S. Candan, L. Xie, and A. Kelliher. 2010. How does the data sampling strategy impact the discovery of information diffusion in social media? *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media (ICWSM)*, 10: 34–41. 144
- F. De la Torre et al. 2009. Guide to the Carnegie Mellon University multimodal activity (CMU-MMAC) database. Technical report. 57

- Y.-A. de Montjoye, E. Shmueli, S. S. Wang, and A. S. Pentland. 2014a. openPDS: Protecting the privacy of metadata through SafeAnswers. *PLoS ONE*, 9(7): e98790. DOI: [10.1371/journal.pone.0098790](https://doi.org/10.1371/journal.pone.0098790). 188
- Y.-A. de Montjoye, A. Stopczynski, E. Shmueli, A. Pentland, and S. Lehmann. 2014b. The strength of the strongest ties in collaborative problem solving. *Scientific Reports*, 4:5277. DOI: [10.1038/srep05277](https://doi.org/10.1038/srep05277). 186
- Y.-A. de Montjoye, L. Radaelli, V. K. Singh, and A. S. Pentland. 2015. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science*, 347(6221): 536–539. DOI: [10.1126/science.1256297](https://doi.org/10.1126/science.1256297). 188
- J. Dean and S. Ghemawat. 2008. MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1): 107–133. DOI: [10.1145/1327452.1327492](https://doi.org/10.1145/1327452.1327492). 279
- A. Delvinioti, H. Jégou, L. Amsaleg, and M. Houle. January 2014. Image retrieval with reciprocal and shared nearest neighbors. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, pp. 321–328. 106
- M. Demirkus, D. Precup, J. J. Clark, and T. Arbel. 2014. Probabilistic temporal head pose estimation using a hierarchical graphical model. In *ECCV*, pp. 328–344. DOI: [10.1007/978-3-319-10590-1_22](https://doi.org/10.1007/978-3-319-10590-1_22). 57
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*. DOI: [10.1109/CVPR.2009.5206848](https://doi.org/10.1109/CVPR.2009.5206848). 9
- S. Deterding, D. Dixon, R. Khaled, and L. Nacke. 2011. From game design elements to gamefulness: Defining gamification. In *Proceedings of ACM International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek)*, pp. 9–15. DOI: [10.1145/2181037.2181040](https://doi.org/10.1145/2181037.2181040). 261
- S. Dey, C. Invited, and P. Paper. 2012. Cloud mobile media: Opportunities, challenges, and directions. In *2012 International Conference on Computing, Networking and Communications, ICNC'12*, pp. 929–933. DOI: [10.1109/ICCNC.2012.6167561](https://doi.org/10.1109/ICCNC.2012.6167561). 290, 291
- D. Dietrich, W. Kastner, T. Maly, C. Roesener, G. Russ, and H. Schweinzer. 2004. Situation modeling. In *Proceedings of the 2004 IEEE International Workshop on Factory Communication Systems*, pp. 93–102. IEEE. DOI: [10.1109/WFCS.2004.1377687](https://doi.org/10.1109/WFCS.2004.1377687). 165, 167
- T. G. Dietterreich, R. H. Lathrop, and T. Lozano-Perez. 1998. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89:31–71. DOI: [10.1007/3-540-45153-6_20](https://doi.org/10.1007/3-540-45153-6_20). 48
- M. V. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. 2010. Fully homomorphic encryption over the integers. In *Advances in Cryptology—EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pp. 24–43. Springer. DOI: [10.1007/978-3-642-13190-5_2](https://doi.org/10.1007/978-3-642-13190-5_2). 101
- T. Dillon, C. Wu, and E. Chang. 2010. Cloud computing: Issues and challenges. In *Proceedings of IEEE International Conference on Advanced Information Networking and Applications (AINA)*, pp. 27–33. DOI: [10.1007/978-3-642-13190-5_2](https://doi.org/10.1007/978-3-642-13190-5_2). 259

- P. Dimandis. 2012. Abundance is our future. In *TED Conference*. 162
- N. Do, C. Hsu, and N. Venkatasubramanian. 2012. CrowdMAC: A crowdsourcing system for mobile access. In *Proceedings of ACM/IFIP/USENIX Middleware*, pp. 1–20. DOI: [10.1007/978-3-642-35170-9_1](https://doi.org/10.1007/978-3-642-35170-9_1). 263, 264, 265
- N. Do, Y. Zhao, C. Hsu, and N. Venkatasubramanian. 2016. Crowdsourced mobile data transfer with delay bound. *ACM Transactions on Internet Technology (TOIT)*, 16(4): 1–29. DOI: [10.1145/2939376](https://doi.org/10.1145/2939376). 263, 264, 265, 267
- T. M. T. Do and D. Gatica-Perez. 2013. Human interaction discovery in smartphone proximity networks. *Personal and Ubiquitous Computing*, 17(3): 413–431. DOI: [10.1007/s00779-011-0489-7](https://doi.org/10.1007/s00779-011-0489-7). 55
- C. Dominguez, M. Vidulich, E. Vogel, and G. McMillan. 1994. Situation awareness: Papers and annotated bibliography. Armstrong Laboratory, Human System Center, ref. Technical report, AL/CF-TR-1994-0085. 166, 167
- J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, and T. Darrell. 2017. Long-term recurrent convolutional networks for visual recognition and description. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4): 677–691 DOI: [10.1109/TPAMI.2016.2599174](https://doi.org/10.1109/TPAMI.2016.2599174). 12, 15, 16, 17, 24, 29
- W. Dong, M. Charikar, and K. Li. July 2008a. Asymmetric distance estimation with sketches for similarity search in high-dimensional spaces. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development (SIGIR)*, pp. 123–130. DOI: [10.1145/1390334.1390358](https://doi.org/10.1145/1390334.1390358). 110, 122, 123
- W. Dong, Z. Wang, W. Josephson, M. Charikar, and K. Li. October 2008b. Modeling LSH for performance tuning. In *Proceedings of the 17th Conference on Information and Knowledge Management (CIKM)*, pp. 669–678. DOI: [10.1145/1458082.1458172](https://doi.org/10.1145/1458082.1458172). 115, 117
- W. Dong, M. Charikar, and K. Li. March 2011. Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*, pp. 577–586. DOI: [10.1145/1963405.1963487](https://doi.org/10.1145/1963405.1963487). 109, 133
- J. J. Dongarra, J. Du Croz, S. Hammarling, and I. S. Duff. 1990. A set of level 3 basic linear algebra subprograms. *ACM Transactions on Mathematical Software (TOMS)*, 16(1): 1–17. DOI: [10.1145/77626.79170](https://doi.org/10.1145/77626.79170). 111
- B. Dostal. 2007. Enhancing situational understanding through employment of unmanned aerial vehicle. *Army Transformation Taking Shape: Interim Brigade Combat Team Newsletter*. 166, 167
- C. Dousson, P. Gaborit, and M. Ghallab. 1993. Situation recognition: Representation and algorithms. In *International Joint Conference on Artificial Intelligence*, volume 13, pp. 166–166. 166, 167
- M. Douze, H. Jégou, H. Singh, L. Amsaleg, and C. Schmid. July 2009. Evaluation of GIST descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, pp. 140–147. 132

- M. Douze, H. Jégou, and F. Perronnin. October 2016. PCIVRolysemous codes. In *ECCV*, pp. 785–801. DOI: [10.1007/978-3-319-46475-6_48](https://doi.org/10.1007/978-3-319-46475-6_48). 109, 134
- J. Downie. 2008. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4): 247–255. DOI: [10.1250/ast.29.247](https://doi.org/10.1250/ast.29.247). 37
- S. Duan, J. Zhang, P. Roe, and M. Towsey. 2014. A survey of tagging techniques for music, speech and environmental sound. *Artificial Intelligence Review*, 42(4): 637–661. DOI: [10.1007/s10462-012-9362-y](https://doi.org/10.1007/s10462-012-9362-y). 38
- L. Dubbeld. 2002. Protecting personal data in camera surveillance practices. *Surveillance & Society*, 2(4). 91
- R. Duda and P. Hart. 1996. *Pattern Classification and Scene Analysis*. Wiley. 167
- F. Dufaux and T. Ebrahimi. 2004. Video surveillance using JPEG 2000. In *International Society for Optics and Photonics. Optical Science and Technology*, pp. 268–275. DOI: [10.1117/12.564828](https://doi.org/10.1117/12.564828). 91
- F. Dufaux and T. Ebrahimi. 2008. Scrambling for privacy protection in video surveillance systems. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(8): 1168–1174. DOI: [10.1109/TCSVT.2008.928225](https://doi.org/10.1109/TCSVT.2008.928225). 92, 93
- N. Eagle and A. Pentland. 2006. Reality mining: Sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4): 255–268. DOI: [10.1007/s00779-005-0046-3](https://doi.org/10.1007/s00779-005-0046-3). 55, 187
- P. Ekman. 1992. An argument for basic emotions. *Cognition and Emotion*, 6(3/4): 169–200. DOI: [10.1080/02699939208411068](https://doi.org/10.1080/02699939208411068). 236
- A. El Ali, A. Matvienko, Y. Feld, W. Heuten, and S. Boll. 2016. VapeTracker: Tracking vapor consumption to help e-cigarette users quit. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 2049–2056. ACM. DOI: [10.1145/2851581.2892318](https://doi.org/10.1145/2851581.2892318). 188
- T. Elgamal. 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Proceedings of the 4th Annual International Cryptology Conference*, volume 196 of *Lecture Notes in Computer Science*, pp. 10–18. Springer. DOI: [10.1109/TIT.1985.1057074](https://doi.org/10.1109/TIT.1985.1057074). 79
- T. Elgamal, M. Yabandeh, A. Aboulmaga, W. Mustafa, and M. Hefeeda. 2015. sPCA: Scalable principal component analysis for big data on distributed platforms. In *Proceedings of ACM International Conference on Management of Data (SIGMOD)*, pp. 79–91. DOI: [10.1145/2723372.2751520](https://doi.org/10.1145/2723372.2751520). 276, 277, 278, 279, 280
- P. Elias and L. Roberts. 1963. *Machine Perception of Three-Dimensional Solids*. PhD thesis, Massachusetts Institute of Technology. <http://hdl.handle.net/1721.1/11589>. 159
- B. Elizalde, G. Friedland, H. Lei, and A. Divakaran. October 2012. There is no data like less data: percepts for video concept detection on consumer-produced media. In *Proceedings of ACM Multimedia 2012 (MM'12)*, pp. 27–32. DOI: [10.1145/2390214.2390223](https://doi.org/10.1145/2390214.2390223). 43

- B. Elizalde, A. Kumar, R. Badlani, A. Bhatnagar, A. Shah, R. Singh, B. Raj, and I. Lane, 2016. Never Ending Learning of Sound. <http://nels.cs.cmu.edu>. 48
- D. P. W. Ellis. March 2007. Beat tracking by dynamic programming. *J. New Music Research*, 36(1): 51–60. <http://www.ee.columbia.edu/~dpwe/pubs/Ellis07-beattrack.pdf>. DOI: 10.1080/09298210701653344. Special Issue on Tempo and Beat Extraction. 38
- D. P. W. Ellis and G. Poliner. 2007. Identifying cover songs with chroma features and dynamic programming beat tracking. In *Proceedings of the ICASSP-07*, pp. 1429–1432. Hawai'i. <http://www.ee.columbia.edu/~dpwe/pubs/EllisP07-coversongs.pdf>. DOI: 10.1109/ICASSP.2007.367348. 38
- D. P. W. Ellis, R. Singh, and S. Sivasdas. 2001. Tandem acoustic modeling in large-vocabulary recognition. In *ICASSP*, pp. 1–517–520. Salt Lake City. <http://www.ee.columbia.edu/~dpwe/pubs/icassp01-spine.pdf>. 36
- P. Embrechts, T. Liniger, and L. Lin. Aug. 2011. Multivariate Hawkes processes: An application to financial data. *Journal of Applied Probability*, pp. 367–378. DOI: 10.1017/S0021900200099344. 200
- M. Endsley. May 1988. Situation awareness global assessment technique (SAGAT). In *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference*, volume 3, pp. 789–795. DOI: 10.1109/NAECON.1988.195097. 165, 167
- Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni, 2006. SPEED project. <http://www.speedproject.eu/>. 76
- Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni. 2007. Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing. *EURASIP Journal on Information Security*, 2007, article 78943. DOI: 10.1155/2007/78943. 77, 79, 80
- Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft. 2009. Privacy-preserving face recognition. In *Proceedings of the 9th International Symposium, Privacy Enhancing Technologies*, volume 5672 of *Lecture Notes in Computer Science*, pp. 235–253. Springer. DOI: 10.1155/2007/78943. 82, 94, 96
- Z. Erkin, M. Beye, T. Veugeri, and R. L. Lagendijk. 2011. Efficiently computing private recommendations. In *ICASSP*, pp. 5864–5867. DOI: 10.1504/IJACT.2014.062738. 82
- S. Escalera, X. Bar, J. Gonzalez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce, H. J. Escalante, J. Shotton, and I. Guyon. 2014. ChaLearn Looking at People Challenge 2014: Dataset and results. In *ECCV Workshops*, pp. 459–473. 57
- N. Evans, S. Marcel, A. Ross, and A. B. J. Teoh. 2015. Biometrics security and privacy protection [from the guest editors]. *IEEE Signal Processing Magazine*, 32(5): 17–18. 83
- M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2012. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. <http://www.pascalnetwork.org/challenges/VOC/voc2012/workshop/index.html>. 227

- M. Everingham, S. M. A. Eslami, L. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2014. The Pascal Visual Object Classes Challenge: A retrospective. *International Journal of Computer Vision (IJCV)*, 111(1): 98–136. [220](#), [226](#)
- H. J. Eysenck. 1947. *Dimensions of Personality*. The International Library of Psychology. Transaction Publishers.
- R. Fagin. June 1998. Fuzzy queries in multimedia database systems. In *Proceedings of the ACM Symposium on Principles of Database Systems*, pp. 1–10. DOI: [10.1145/275487.275488](#). [105](#)
- R. Fagin, R. Kumar, and D. Sivakumar. 2003. Efficient similarity search and classification via rank aggregation. In *SIGMOD*, pp. 301–312. DOI: [10.1145/872757.872795](#). [114](#)
- C.-I. Fan, S.-Y. Huang, and W.-C. Hsu. 2015. Encrypted data deduplication in cloud storage. In *2015 10th Asia Joint Conference on Information Security (AsiaJCS)*, pp. 18–25. IEEE. DOI: [10.1109/AsiaJCS.2015.12](#). [94](#)
- Farach-Colton and P. Indyk. October 1999. Approximate nearest neighbor algorithms for Hausdorff metrics via embeddings. In *Proceedings of the Symposium on the Foundations of Computer Science*, pp. 171–179. DOI: [10.1109/SFFCS.1999.814589](#). [133](#)
- C. Feichtenhofer, A. Pinz, and A. Zisserman. 2016. Convolutional two-stream network fusion for video action recognition. In *CVPR*, pp. 1933–1941. [11](#), [24](#)
- P. Flajolet and G. N. Martin. October 1985. Probabilistic counting algorithms for data base applications. *Journal of Computer and System Sciences*, 31(2): 182–209. DOI: [10.1016/0022-0000\(85\)90041-8](#). [120](#)
- M. Flickner, S. Harpreet, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. September 1995. Query by image and video content: The QBIC system. *Computer*, 28: 23–32. DOI: [10.1109/2.410146](#). [105](#)
- C. Fontaine and F. Galand. 2007. A survey of homomorphic encryption for nonspecialists. *Hindawi Publishing Corporation, EURASIP Journal on Information Security*, 2007, pp. 1–10. DOI: [10.1155/2007/13801](#). [77](#)
- J. Fortmann, E. Root, S. Boll, and W. Heuten. 2016. Tangible apps bracelet: Designing modular wrist-worn digital jewellery for multiple purposes. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems, DIS '16*, pp. 841–852. ACM, New York. <http://doi.acm.org/10.1145/2901790.2901838>. DOI: [10.1145/2901790.2901838](#). [188](#)
- T. Foulsham, R. Dewhurst, M. Nyström, H. Jarodzka, R. Johansson, G. Underwood, and K. Holmqvist. 2012. Comparing scanpaths during scene encoding and recognition: A multi-dimensional approach. *Journal of Eye Movement Research*, 5(4:3): 1–14. DOI: [10.16910/jemr.5.4.3](#). [229](#)
- M. Franz and S. Katzenbeisser. 2011. Processing encrypted floating point signals. In *Proceedings of the Thirteenth ACM Multimedia Workshop on Multimedia and Security*, pp. 103–108. ACM. DOI: [10.1145/2037252.2037271](#). [102](#)

- M. Franz, B. Deiseroth, K. Hamacher, S. Katzenbeisser, S. Jha, and H. Schroeder. 2010. Secure computations on real-valued signals. In *Proceedings of the IEEE Workshop on Information Forensics and Security (WIFS)*, pp. 25–27. DOI: [10.1109/WIFS.2010.5711458](https://doi.org/10.1109/WIFS.2010.5711458). 102
- G. Friedland and D. van Leeuwen. 2010. Speaker recognition and diarization. In *Semantic Computing*, pp. 115–130. IEEE Press/Wileys. DOI: [10.1002/9780470588222.ch7](https://doi.org/10.1002/9780470588222.ch7). 36
- J. H. Friedman, J. L. Bentley, and R. A. Finkel. 1977. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3): 209–226. DOI: [10.1145/355744.355745](https://doi.org/10.1145/355744.355745). 107
- J.-J. Fuchs. November 2011. Spread representations. In *ASLOMAR*. DOI: [10.1109/ACSSC.2011.6190120](https://doi.org/10.1109/ACSSC.2011.6190120). 126
- T. Fujishima. 1999. Realtime chord recognition of musical sound: A system using Common Lisp music. In *Proceedings International Computer Music Conference*, pp. 464–467. http://www-ccrma.stanford.edu/~jos/mus423h/Real_Time_Chord_Recognition_Musical.html. 37
- K. Fukushima. 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4): 193–202. 5
- T. Furon and H. Jégou. February 2013. Using extreme value theory for image detection. Research Report RR-8244, INRIA. 106
- T. Furon, H. Jégou, L. Amsaleg, and B. Mathon. November 2013. Fast and secure similarity search in high dimensional space. In *WIFS*, pp. 73–78. DOI: [10.1109/WIFS.2013.6707797](https://doi.org/10.1109/WIFS.2013.6707797). 105
- GaiKai. January 2015. GaiKai web page. <http://www.gaikai.com/>. 287, 291
- GamingAnywhere Repository. 2013. GamingAnywhere: An open source cloud gaming project. <http://gaminganywhere.org>. 292, 294, 296
- R. Ganti, F. Ye, and H. Lei. 2011. Mobile crowdsensing: Current state and future challenges. *IEEE Communication Magazine*, 49(11): 32–39. DOI: [10.1109/MCOM.2011.6069707](https://doi.org/10.1109/MCOM.2011.6069707). 268
- M. Gao. 2012. *EventShop: A scalable framework for analysis of spatio-temporal-thematic data streams*. PhD thesis, University of California, Irvine. 183
- M. Gao, V. K. Singh, and R. Jain. 2012. EventShop: From heterogeneous web streams to personalized situation detection and control. In *Proceedings of the 4th Annual ACM Web Science Conference*, pp. 105–108. ACM. DOI: [10.1145/2380718.2380733](https://doi.org/10.1145/2380718.2380733). 161, 177, 185
- L. P. García-Perera, J. A. Nolzco-Flores, B. Raj, and R. M. Stern. 2012. Optimization of the DET curve in speaker verification. In *2012 IEEE Spoken Language Technology Workshop (SLT), Miami, FL, USA, December 2–5, 2012*, pp. 318–323. DOI: [10.1109/SLT.2012.6424243](https://doi.org/10.1109/SLT.2012.6424243). 36

- L. P. García-Perera, B. Raj, and J. A. Nolzco-Flores. 2013a. Ensemble approach in speaker verification. In *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25–29, 2013*, pp. 2455–2459. 36
- L. P. García-Perera, B. Raj, and J. A. Nolzco-Flores. 2013b. Optimization of the DET curve in speaker verification under noisy conditions. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26–31, 2013*, pp. 7765–7769. 36
- D. Gatica-Perez. 2009. Automatic nonverbal analysis of social interaction in small groups: A review. *Image and Vision Computing*, 27(12): 1775–1787. DOI: [10.1016/j.imavis.2009.01.004](https://doi.org/10.1016/j.imavis.2009.01.004). 55
- T. Ge, K. He, Q. Ke, and J. Sun. June 2013. Optimized product quantization for approximate nearest neighbor search. In *CVPR*, pp. 744–755. DOI: [10.1109/TPAMI.2013.240](https://doi.org/10.1109/TPAMI.2013.240). 131
- I.-D. Gebru, X. Alameda-Pineda, F. Forbes, and R. Horaud. 2016. EM algorithms for weighted-data clustering with application to audio-visual scene analysis. *IEEE TPAMI*, 38(12): 2402–2415. DOI: [10.1109/TPAMI.2016.2522425](https://doi.org/10.1109/TPAMI.2016.2522425). 55
- H. W. Gellersen, A. Schmidt, and M. Beigl. 2002. Multi-sensor context-awareness in mobile devices and smart artifacts. *Mobile Networks and Applications*, 7(5): 341–351. DOI: [10.1023/A:1016587515822](https://doi.org/10.1023/A:1016587515822). 187
- X. Geng and Y. Xia. 2014. Head pose estimation based on multivariate label distribution. In *CVPR*. DOI: [10.1109/CVPR.2014.237](https://doi.org/10.1109/CVPR.2014.237). 57
- C. Gentry. 2009. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University. 77, 101
- L. Georgiadis, M. Neely, and L. Tassiulas. 2006. Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networking*. DOI: [10.1561/1300000001](https://doi.org/10.1561/1300000001). 265
- I. Ghosh and V. Singh. 2016. Predicting privacy attitudes using phone metadata. In *Proceedings of the International Conference on Social Computing, Behavioral-Cultural Modeling & Prediction and Behavior Representation in Modeling and Simulation, Washington DC*, volume 1, pp. 51–60. DOI: [10.1007/978-3-319-39931-7_6](https://doi.org/10.1007/978-3-319-39931-7_6). 188
- D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M. Plumbley, 2013. IEEE AASP challenge: Detection and classification of acoustic scenes and events. Web resource, available: <http://c4dm.eecs.qmul.ac.uk/sceneseventschallenge/>. DOI: [10.1109/WASPAA.2013.6701819](https://doi.org/10.1109/WASPAA.2013.6701819). 38
- S. O. Gilani, R. Subramanian, Y. Yan, D. Melcher, N. Sebe, and S. Winkler. 2015. PET: An eye-tracking dataset for animal-centric Pascal object classes. In *International Conference on Multimedia & Expo (ICME)*. DOI: [10.1109/ICME.2015.7177450](https://doi.org/10.1109/ICME.2015.7177450). 221, 226, 230, 233, 234, 236
- A. Gionis, P. Indyk, and R. Motwani. 1999. Similarity search in high dimension via hashing. In *Proceedings of the International Conference on Very Large Databases*, pp. 518–529. 105, 113, 132

- M. Girolami, S. Lenzi, F. Furfari, and S. Chessa. 2008. SAIL: A sensor abstraction and integration layer for context awareness. In *2008 34th Euromicro Conference Software Engineering and Advanced Applications*, pp. 374–381. IEEE. DOI: [10.1109/SEAA.2008.30](https://doi.org/10.1109/SEAA.2008.30). 187
- R. Girshick. 2015. Fast R-CNN. In *ICCV*, pp. 1440–1448. DOI: [10.1109/ICCV.2015.169](https://doi.org/10.1109/ICCV.2015.169). 3, 9
- R. Girshick, J. Donahue, T. Darrell, and J. Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pp. 580–587. DOI: [10.1109/CVPR.2014.81](https://doi.org/10.1109/CVPR.2014.81). 9
- A. Goldberg, B. Recht, J. Xu, R. Nowak, and X. Zhu. 2010. Transduction with matrix completion: Three birds with one stone. In *NIPS*, pp. 757–765. 57, 60, 62, 68
- D. Goldman. Apr. 2012. Google unveils ‘Project Glass’ virtual-reality glasses. In *Money*. DOI: [10.1109/NWeSP.2011.6088206](https://doi.org/10.1109/NWeSP.2011.6088206). 313
- O. Goldreich, S. Micali, and A. Wigderson. 1987. How to play any mental game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, pp. 218–229. ACM. DOI: [10.1145/28395.28420](https://doi.org/10.1145/28395.28420). 80
- S. Goldwasser and S. Micali. 1984. Probabilistic encryption. *Elsevier Journal of Computer and System Sciences*, 28(2): 270–299. DOI: [10.1016/0022-0000\(84\)90070-9](https://doi.org/10.1016/0022-0000(84)90070-9). 80
- B. Golub and M. O. Jackson. 2010. Using selection bias to explain the observed structure of internet diffusions. *Proceedings of the National Academy of Sciences*, 107(24): 10833–10836. DOI: [10.1073/pnas.1000814107](https://doi.org/10.1073/pnas.1000814107). 145
- Y. Gong and S. Lazebnik. June 2011. Iterative quantization: A procrustean approach to learning binary codes. In *CVPR*, pp. 817–824. DOI: [10.1109/CVPR.2011.5995432](https://doi.org/10.1109/CVPR.2011.5995432). 122, 124
- Z. Gong, X. Gu, and J. Wilkes. 2010. PRESS: Predictive elastic resource scaling for cloud systems. In *Proceedings of IEEE International Conference on Network and Service Management (CNSM)*, pp. 9–16. DOI: [10.1109/CNSM.2010.5691343](https://doi.org/10.1109/CNSM.2010.5691343). 259
- M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. 2008. Understanding individual human mobility patterns. *Nature*, 453(7196): 779–782. DOI: [10.1038/nature06958](https://doi.org/10.1038/nature06958). 187
- I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio. 2013. Maxout networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1319–1327. 6
- D. Gorisse, M. Cord, and F. Precioso. February 2012. Locality-sensitive hashing for CHI2 distance. *IEEE Trans. PAMI*, 34(2): 402–409. DOI: [10.1109/TPAMI.2011.193](https://doi.org/10.1109/TPAMI.2011.193). 133
- G. J. Gorn. 1982. The effects of music in advertising on choice behavior: A classical conditioning approach. *The Journal of Marketing*, pp. 94–101. DOI: [10.2307/1251163](https://doi.org/10.2307/1251163). 187
- M. Goto. 2001. A predominant-F0 estimation method for CD recordings: Map estimation using EM algorithm for adaptive tone models. In *Proceedings ICASSP-2001*, pp. 3365–3368. 37

- V. K. Goyal, M. Vetterli, and N. T. Thao. January 1998. Quantized overcomplete expansions in \mathcal{R}^N : Analysis, synthesis, and algorithms. *IEEE Trans. Inform. Theory*, 44(1): 16–31. DOI: [10.1109/18.650985](https://doi.org/10.1109/18.650985). 121, 124
- K. Graffi, D. Stingl, C. Gross, H. Nguyen, A. Kovacevic, and R. Steinmetz. 2010. Towards a P2P cloud: Reliable resource reservations in unreliable P2P systems. In *Proceedings of IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 27–34. DOI: [10.1109/ICPADS.2010.34](https://doi.org/10.1109/ICPADS.2010.34). 259
- A. Graves. 2012. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer. 6
- A. Graves, A. Mohamed, and G. E. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *ICASSP*, pp. 6645–6649. DOI: [10.1109/ICASSP.2013.6638947](https://doi.org/10.1109/ICASSP.2013.6638947). 3, 12
- R. M. Gray and D. L. Neuhoff. October 1998. Quantization. *IEEE Transactions on Information Theory*, 44: 2325–2384. DOI: [10.1109/18.720541](https://doi.org/10.1109/18.720541). 117, 127, 128
- J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, and G. Wang. 2016. Recent advances in convolutional neural networks. In *arXiv:1512.07108*. 6
- S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. 2013. YouTube2Text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*, pp. 2712–2719. DOI: [10.1109/ICCV.2013.337](https://doi.org/10.1109/ICCV.2013.337). 16, 26
- J. Guo, C. Gurrin, and S. Lao. 2013. Who produced this video, amateur or professional? In *Proceedings of the 3rd ACM International Conference on Multimedia Retrieval*, pp. 271–278. ACM. DOI: [10.1145/2461466.2461509](https://doi.org/10.1145/2461466.2461509). 147
- S. Gupta and J. Nicholson. 1985. Simple visual reaction time, personality strength of the nervous system: Theory approach. *Personality and Individual Differences*, 6(4): 461–469. DOI: [10.1016/0191-8869\(85\)90139-4](https://doi.org/10.1016/0191-8869(85)90139-4). 244
- A. Guttman. 1984. R-trees: A dynamic index structure for spatial searching. In *SIGMOD*, pp. 47–57. DOI: [10.1145/602259.602266](https://doi.org/10.1145/602259.602266). 107
- H. Haddadi, H. Howard, A. Chaudhry, J. Crowcroft, A. Madhavapeddy, and R. Mortier. 2015. Personal data: Thinking inside the box. In *arXiv:1501.04737*. 188
- A. Hanjalic. 2013. Multimedia retrieval that matters. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 9(1s): 44. DOI: [10.1145/2490827](https://doi.org/10.1145/2490827). 155
- A. Hanjalic and L.-Q. Xu. 2005. Affective video content representation and modeling. *IEEE Transactions on Multimedia*, 7(1): 143–154. DOI: [10.1109/TMM.2004.840618](https://doi.org/10.1109/TMM.2004.840618). 221
- A. Hanjalic, C. Kofler, and M. Larson. 2012. Intent and its discontents: The user at the wheel of the online video search engine. In *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 1239–1248. ACM. DOI: [10.1145/2393347.2396424](https://doi.org/10.1145/2393347.2396424). 140
- D. A. Harville. 1998. *Matrix Algebra from a Statistician's Perspective*. Springer-verlag. 185
- A. G. Hawkes. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, pp. 83–90. DOI: [10.1093/biomet/58.1.83](https://doi.org/10.1093/biomet/58.1.83). 193, 197, 198, 200

- A. G. Hawkes and D. Oakes. 1974. A cluster process representation of a self-exciting process. *Journal of Applied Probability*, 11: 493–503. DOI: [10.2307/3212693](https://doi.org/10.2307/3212693). 200
- S. Haynes and R. Jain. 1986. Event detection and correspondence. *Optical Engineering*, 25: 387–393. DOI: [10.1117/12.7973835](https://doi.org/10.1117/12.7973835). 159
- K. He, F. Wen, and J. Sun. June 2013. K-means hashing: An affinity-preserving quantization method for learning binary compact codes. In *CVPR*, pp. 2938–2945. DOI: [10.1109/CVPR.2013.378](https://doi.org/10.1109/CVPR.2013.378). 121
- K. He, X. Zhang, S. Ren, and J. Sun. 2016a. Identity mappings in deep residual networks. In *ECCV*, pp. 630–645. DOI: [10.1007/978-3-319-46493-0_38](https://doi.org/10.1007/978-3-319-46493-0_38). 6
- K. He, X. Zhang, S. Ren, and J. Sun. 2016b. Deep residual learning for image recognition. In *CVPR*, pp. 770–778. DOI: [10.1109/cvpr.2016.90](https://doi.org/10.1109/cvpr.2016.90). 5, 6, 9
- L. He, G. Liu, and C. Yuchen. Jul. 2014. Buffer status and content aware scheduling scheme for cloud gaming based on video streaming. In *2014 IEEE International Conference on Multimedia and Expo Workshops (ICME)*, pp. 1–6. DOI: [10.1109/ICMEW.2014.6890629](https://doi.org/10.1109/ICMEW.2014.6890629). 309
- F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. 2015. ActivityNet: A large-scale video benchmark for human activity understanding. In *CVPR*, pp. 1914–1923. 19, 22, 23
- M. Hemmati, A. Javadtalab, A. A. Nazari Shirehjini, S. Shirmohammadi, and T. Arici. 2013. Game as video: Bit rate reduction through adaptive object encoding. In *Proceedings of the 23rd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video, NOSSDAV '13*, pp. 7–12. ACM, New York. DOI: [10.1145/2460782.2460784](https://doi.org/10.1145/2460782.2460784). 310
- J. Hershey, S. Rennie, P. Olsen, and T. Kristjansson. 2010. Super-human multi-talker speech recognition: A graphical modeling approach. *Computer Speech & Language*, 24(1): 45–66. DOI: [10.1016/j.csl.2008.11.001](https://doi.org/10.1016/j.csl.2008.11.001). 37
- G. Higgins. December 14, 1993. System for distributing, processing and displaying financial information. US Patent 5,270,922. 165
- G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, pp. 82–97. DOI: [10.1109/MSP.2012.2205597](https://doi.org/10.1109/MSP.2012.2205597). 3
- G. E. Hinton, S. Osindero, and Y.-W. Teh. 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7): 1527–1554. DOI: [10.1162/neco.2006.18.7.1527](https://doi.org/10.1162/neco.2006.18.7.1527).
- M. Hirt and K. Sako. 2000. Efficient receipt-free voting based on homomorphic encryption. In *Advances in Cryptology*, volume 1807 of *Lecture Notes in Computer Science*, pp. 539–556. Springer. DOI: [10.1007/3-540-45539-6_38](https://doi.org/10.1007/3-540-45539-6_38). 81
- G. Hjaltason and H. Samet. 1998. Incremental distance join algorithms for spatial databases. In *ACM SIGMOD Record*, volume 27, pp. 237–248. ACM. DOI: [10.1145/276305.276326](https://doi.org/10.1145/276305.276326). 173

- S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8): 1735–1780. DOI: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735). 16
- S. C. Hoi, W. Liu, and S.-F. Chang. 2008. Semi-supervised distance metric learning for collaborative image retrieval. In *CVPR*, pp. 1–7. DOI: [10.1145/1823746.1823752](https://doi.org/10.1145/1823746.1823752). 149
- H. Hong, D. Chen, C. Huang, K. Chen, and C. Hsu. 2015. Placing virtual machine to optimize cloud gaming experience. *IEEE Transactions on Cloud Computing*, 3(1): 42–53. DOI: [10.1109/TCC.2014.2338295](https://doi.org/10.1109/TCC.2014.2338295). 259
- H. Hong, J. Chuang, and C. Hsu. 2016a. Animation rendering on multimedia fog computing platforms. In *Proceedings of IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 336–343. DOI: [10.1109/CloudCom.2016.0060.274](https://doi.org/10.1109/CloudCom.2016.0060.274), 275
- H. Hong, P. Tsai, and C. Hsu. 2016b. Dynamic module deployment in a fog computing platform. In *Proceedings of the Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp. 1–6. DOI: [10.1109/APNOMS.2016.7737202](https://doi.org/10.1109/APNOMS.2016.7737202). 282, 283, 284
- H.-J. Hong, D.-Y. Chen, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu. 2014a. Placing virtual machines to optimize cloud gaming experience. *IEEE Transactions on Cloud Computing*, pp. 1–2. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6853364>. DOI: [10.1109/TCC.2014.2338295](https://doi.org/10.1109/TCC.2014.2338295). 296
- H.-J. Hong, T.-Y. Fan-Chiang, C.-R. Lee, K.-T. Chen, C.-Y. Huang, and C.-H. Hsu. 2014b. GPU consolidation for cloud games: Are we there yet? *2014 13th Annual Workshop on Network and Systems Support for Games (NetGames)*, article 3. DOI: [10.1109/NetGames.2014.7008969](https://doi.org/10.1109/NetGames.2014.7008969). 297, 302
- C.-F. Hsu, T.-H. Tsai, C.-Y. Huang, C.-H. Hsu, and K.-T. Chen. Mar 2015. Screenshot dissected: Performance measurements and design considerations. In *Proceedings of ACM Multimedia Systems 2015*, pp. 177–188. DOI: [10.1145/2713168.2713176](https://doi.org/10.1145/2713168.2713176). 297
- C. Y. Hsu, C. S. Lu, and S. C. Pei. 2009. Secure and robust SIFT. In *Proceedings of the 17th ACM International Conference on Multimedia*, pp. 637–640. DOI: [10.1145/1631272.1631376](https://doi.org/10.1145/1631272.1631376). 86, 87
- C. Y. Hsu, C. S. Lu, and S. C. Pei. 2011. Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction. *Proceedings of the International Society for Optics and Photonics, IS&T/SPIE Electronic Imaging*, 7880(788005). DOI: [10.1117/12.873325](https://doi.org/10.1117/12.873325). 86, 87, 90
- G. Hu and D. L. Wang. 2003. Monaural speech separation. In *NIPS*, volume 13. MIT Press, Cambridge MA. 37
- X.-S. Hua, L. Yang, J. Wang, J. Wang, M. Ye, K. Wang, Y. Rui, and J. Li. 2013. Clickage: Towards bridging semantic and intent gaps via mining click logs of search engines. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 243–252. ACM. DOI: [10.1145/2502081.2502283](https://doi.org/10.1145/2502081.2502283). 148
- C.-Y. Huang, D.-Y. Chen, C.-H. Hsu, and K.-T. Chen. Oct. 2013a. GamingAnywhere: An open-source cloud gaming testbed. In *Proceedings of ACM Multimedia 2013 (Open Source Software Competition Track)*, pp. 36–47. DOI: [10.1145/2502081.2502222](https://doi.org/10.1145/2502081.2502222). 297

- C.-Y. Huang, C.-H. Hsu, D.-Y. Chen, and K.-T. Chen. 2013b. Quantifying user satisfaction in mobile cloud games. In *Proceedings of Workshop on Mobile Video Delivery, MoViD'14*, pp. 4:1–6. ACM, New York. DOI: [10.1145/2579465.2579468](https://doi.org/10.1145/2579465.2579468). 297
- C.-Y. Huang, K.-T. Chen, D.-Y. Chen, H.-J. Hsu, and C.-H. Hsu. Jan. 2014a. GamingAnywhere: The first open source cloud gaming system. *ACM Transactions on Multimedia Computer Communication Applications*, 10(1s): 10:1–25. DOI: [10.1145/2537855](https://doi.org/10.1145/2537855). 292, 293
- C.-Y. Huang, P.-H. Chen, Y.-L. Huang, K.-T. Chen, and C.-H. Hsu. 2014b. Measuring the client performance and energy consumption in mobile cloud gaming. *2014 13th Annual Workshop on Network and Systems Support for Games (NetGames)*, pp. 4–6. 297
- R. W. Hubbard, S. Magotiaux, and M. Sullivan. 2004. The state use of closed circuit TV: Is there a reasonable expectation of privacy in public? *Crim. LQ*, 49: 222. 91
- D. H. Hubel and T. N. Wiesel. 1968. Receptive fields and functional architecture of monkey striate cortex. *The Journal of Physiology*, pp. 215–243. DOI: [10.1113/jphysiol.1968.sp008455](https://doi.org/10.1113/jphysiol.1968.sp008455). 5
- H. Hung and B. Kröse. 2011. Detecting F-formations as dominant sets. In *Proceedings of the 13th International Conference on Multimodal Interfaces (ICMI)*, pp. 231–238. DOI: [10.1145/2070481.2070525](https://doi.org/10.1145/2070481.2070525). 70
- ICSI, U. Berkeley, Yahoo, and LLNL. 2015. The Multimedia Commons Project. <http://mmcommons.org>. 39
- H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah. 2016. The THUMOS challenge on action recognition for videos “in the wild.” In *arXiv:1604.06182*. DOI: [10.1016/j.cviu.2016.10.018](https://doi.org/10.1016/j.cviu.2016.10.018). 23
- D. Imseng and G. Friedland. 2010. Tuning-robust initialization methods for speaker diarization. *Transactions on Audio, Speech, and Language Processing*, 18(8): 2028–2037. DOI: [10.1109/TASL.2010.2040796](https://doi.org/10.1109/TASL.2010.2040796). 36
- P. Indyk. 2002. Approximate nearest neighbor algorithms for Frechet distance via product metrics. In *Proceedings of the Eighteenth Annual Symposium on Computational Geometry*, pp. 102–106. DOI: [10.1145/513400.513414](https://doi.org/10.1145/513400.513414). 133
- P. Indyk and R. Motwani. 1998. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *STOC*, pp. 604–613. DOI: [10.1145/276698.276876](https://doi.org/10.1145/276698.276876). 105, 107, 108, 114
- P. Indyk and A. Naor. Aug. 2007. Nearest-neighbor-preserving embeddings. *ACM Transactions on Algorithms (TALG)*, 3: article 31. DOI: [10.1145/1273340.1273347](https://doi.org/10.1145/1273340.1273347). 133
- P. Indyk and N. Thaper. Oct. 2003. Fast image retrieval via embeddings. In *International Workshop on Statistical and Computational Theories of Vision, ICCV Workshop*. 120
- Infinity Research. Dec. 2016. Global Cloud Gaming Market 2017–2021. <http://www.technavio.com/report/global-gaming-global-cloud-gaming-market-2017-2021>. 287
- S. Ioffe and C. Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pp. 448–456. 6

- R. Iqbal, S. Shirmohammadi, and A. E. Saddik. 2006. Compressed-domain encryption of adapted H.264 video. *Proceedings of IEEE 8th International Symposium on Multimedia*, pp. 979–984. DOI: [10.1109/ISM.2006.50](https://doi.org/10.1109/ISM.2006.50). 92, 93
- N. Islam, W. Puech, and R. Brouzet. 2009. A homomorphic method for sharing secret images. In *Proceedings of the 8th International Workshop on Digital Watermarking*, volume 5703 of *Lecture Notes in Computer Science*, pp. 121–135. Springer. DOI: [10.1007/978-3-642-03688-0_13](https://doi.org/10.1007/978-3-642-03688-0_13). 88
- Y. Isoda, S. Kurakake, and H. Nakano. 2004. Ubiquitous sensors based human behavior modeling and recognition using a spatio-temporal representation of user states. In *18th International Conference on Advanced Information Networking and Applications*, volume 1, pp. 512–517. IEEE. DOI: [10.1109/AINA.2004.1283961](https://doi.org/10.1109/AINA.2004.1283961). 187
- L. Itti and C. Koch. 2000. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10): 1489–1506. DOI: [10.1016/S0042-6989\(99\)00163-7](https://doi.org/10.1016/S0042-6989(99)00163-7). 222
- L. Itti, C. Koch, and E. Niebur. 1998. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. PAMI*, 20(11): 1254–1259. DOI: [10.1109/34.730558](https://doi.org/10.1109/34.730558). 231
- H. Jain, P. Pérez, R. Gribonval, J. Zepeda, and H. Jégou. October 2016. Approximate search with quantized sparse representations. In *ECCV*, pp. 681–696. DOI: [10.1007/978-3-319-46478-7_42](https://doi.org/10.1007/978-3-319-46478-7_42). 131
- R. Jain and D. Sonnen. 2011. Social life networks. *IT Professional*, 13(5): 8–11. DOI: [10.1109/MITP.2011.86](https://doi.org/10.1109/MITP.2011.86). 137
- G. Jakobson, J. Buford, and L. Lewis. Oct. 2006. A framework of cognitive situation modeling and recognition. In *Military Communications Conference, 2006. IEEE*, pp. 1–7. DOI: [10.1109/MILCOM.2006.302076](https://doi.org/10.1109/MILCOM.2006.302076). 166
- M. Jamali and M. Ester. 2009. TrustWalker: A random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 397–406. ACM. DOI: [10.1145/1557019.1557067](https://doi.org/10.1145/1557019.1557067). 152
- B. J. Jansen. 2006. Search log analysis: What it is, what’s been done, how to do it. *Library & Information Science Research*, 28(3): 407–432. DOI: [10.1016/j.lisr.2006.06.005](https://doi.org/10.1016/j.lisr.2006.06.005). 148
- S. Jarvinen, J.-P. Laulajainen, T. Sutinen, and S. Sallinen. 2006. QoS-aware real-time video encoding how to improve the user experience of a gaming-on-demand service. *2006 3rd IEEE Consumer Communications and Networking Conference*, 2: 994–997. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1593187>. DOI [10.1109/CCNC.2006.1593187](https://doi.org/10.1109/CCNC.2006.1593187). 308
- E. Jeannot, C. Kelly, and D. Thompson. 2003. The development of situation awareness measures in ATM systems. EATMP Report HRS/HSP-005-REP-01. 165, 167
- H. Jégou, H. Harzallah, and C. Schmid. June 2007. A contextual dissimilarity measure for accurate and efficient image search. In *CVPR*, pp. 1–8. DOI: [10.1109/CVPR.2007.382970](https://doi.org/10.1109/CVPR.2007.382970). 119

- H. Jégou, L. Amsaleg, C. Schmid, and P. Gros. April 2008a. Query-adaptive locality sensitive hashing. In *ICASSP*, pp. 825–828. DOI: [10.1109/ICASSP.2008.4517737](https://doi.org/10.1109/ICASSP.2008.4517737). 115, 117, 119
- H. Jégou, M. Douze, and C. Schmid. October 2008b. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, pp. 304–317. DOI: [10.1007/978-3-540-88682-2_24](https://doi.org/10.1007/978-3-540-88682-2_24). 120, 123, 132
- H. Jégou, M. Douze, and C. Schmid. February 2010a. Improving bag-of-features for large scale image search. *IJCV*, 87(3): 316–336. DOI: [10.1007/s11263-009-0285-2](https://doi.org/10.1007/s11263-009-0285-2). 116, 119
- H. Jégou, M. Douze, C. Schmid, and P. Pérez. June 2010b. Aggregating local descriptors into a compact image representation. In *CVPR*, pp. 3304–3311. DOI: [10.1109/CVPR.2010.5540039](https://doi.org/10.1109/CVPR.2010.5540039). 9, 124, 131, 132
- H. Jégou, C. Schmid, H. Harzallah, and J. Verbeek. January 2010. Accurate image search using the contextual dissimilarity measure. *IEEE Trans. PAMI*, 32(1): 2–11. DOI: [10.1109/TPAMI.2008.285](https://doi.org/10.1109/TPAMI.2008.285). 106
- H. Jégou, M. Douze, and C. Schmid. January 2011. Product quantization for nearest neighbor search. *IEEE Trans. PAMI*, 33(1): 117–128. DOI: [10.1109/TPAMI.2010.57](https://doi.org/10.1109/TPAMI.2010.57). 106, 110, 112, 123, 127, 128, 129, 130, 131
- H. Jégou, R. Tavenard, M. Douze, and L. Amsaleg. May 2011. Searching in one billion vectors: Re-rank with source coding. In *ICASSP*, pp. 861–864. DOI: [10.1109/ICASSP.2011.5946540](https://doi.org/10.1109/ICASSP.2011.5946540). 131
- H. Jégou, T. Furon, and J.-J. Fuchs. January 2012. Anti-sparse coding for approximate nearest neighbor search. In *ICASSP*, pp. 2029–2032. DOI: [10.1109/ICASSP.2012.6288307](https://doi.org/10.1109/ICASSP.2012.6288307). 123, 124, 127
- J. H. Jensen, M. G. Christensen, D. P. W. Ellis, and S. H. Jensen. May 2009. Quantitative analysis of a common audio similarity measure. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4): 693–703. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4804953&isnumber=4787206>. DOI: [10.1109/TASL.2008.2012314](https://doi.org/10.1109/TASL.2008.2012314). 38
- J. Ji, J. Li, S. Yan, B. Zhang, and Q. Tian. Dec. 2012. Super-bit locality-sensitive hashing. In *NIPS*, pp. 108–116. 124
- S. Ji, W. Xu, M. Yang, and K. Yu. 2010. 3D convolutional neural networks for human action recognition. In *ICML*, pp. 221–231. DOI: [10.1109/TPAMI.2012.59](https://doi.org/10.1109/TPAMI.2012.59). 9, 10
- M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang. 2012a. Social contextual recommendation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pp. 45–54. ACM. DOI: [10.1145/2396761.2396771](https://doi.org/10.1145/2396761.2396771). 147, 152
- M. Jiang, P. Cui, F. Wang, Q. Yang, W. Zhu, and S. Yang. 2012b. Social recommendation across multiple relational domains. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pp. 1422–1431. ACM. DOI: [10.1145/2396761.2398448](https://doi.org/10.1145/2396761.2398448). 152

- M. Jiang, P. Cui, F. Wang, W. Zhu, and S. Yang. 2014a. Scalable recommendation with social contextual information. *IEEE Transactions on Knowledge and Data Engineering*, 26(11): 2789–2802. DOI: [10.1109/TKDE.2014.2300487](https://doi.org/10.1109/TKDE.2014.2300487). 152
- Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. 2011. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *ACM International Conference on Multimedia Retrieval (ICMR)*, p. 29. DOI: [10.1145/1991996.1992025](https://doi.org/10.1145/1991996.1992025). 21
- Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. 2013. High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval (IJMIR)*, pp. 73–101. DOI: [10.1007/s13735-012-0024-2](https://doi.org/10.1007/s13735-012-0024-2). 4
- Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. 2014b. THUMOS challenge: Action recognition with a large number of classes. <http://crcv.ucf.edu/THUMOS14/>. 19, 21
- Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang. 2015. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *arXiv:1502.07209*. 19, 22
- X. Jin, A. Gallagher, L. Cao, J. Luo, and J. Han. 2010. The wisdom of social multimedia: Using Flickr for prediction and forecast. In *Proceedings of the International Conference on Multimedia*, pp. 1235–1244. ACM. DOI: [10.1145/1873951.1874196](https://doi.org/10.1145/1873951.1874196). 156
- Z. Jin, Y. Hu, Y. Lin, D. Zhang, S. Lin, D. Cai, and X. Li. Dec. 2013. Complementary projection hashing. In *ICCV*, pp. 257–264. DOI: [10.1109/ICCV.2013.39](https://doi.org/10.1109/ICCV.2013.39). 118
- T. Joachims. 1999. Transductive inference for text classification using SVM. In *ICML*, pp. 200–209. 68, 70, 71
- J. Johnson, M. Douze, and H. Jégou. Feb. 2017. Billion-scale similarity search with GPU. *arXiv:1702.08734*. 132
- W. B. Johnson and J. Lindenstrauss. 1984. Extensions of Lipschitz mappings into a Hilbert space. *Contemp. Math.*, (26): 189–206. DOI: [10.1090/conm/026/737400](https://doi.org/10.1090/conm/026/737400). 107
- H. Joho, J. Staiano, N. Sebe, and J. M. Jose. 2011. Looking at the viewer: Analysing facial activity to detect personal highlights of multimedia contents. *Multimedia Tools and Applications*, 51(2): 505–523. DOI: [10.1007/s11042-010-0632-x](https://doi.org/10.1007/s11042-010-0632-x). 221, 242, 243
- A. Joly and O. Buisson. 2008. A posteriori multi-probe locality sensitive hashing. In *ACM Multimedia*, pp. 209–218. DOI: [10.1145/1459359.1459388](https://doi.org/10.1145/1459359.1459388). 118
- A. Joly and O. Buisson. Oct. 2009. Logo retrieval with a contrario visual query expansion. In *ACM Multimedia*, pp. 581–584. DOI: [10.1145/1631272.1631361](https://doi.org/10.1145/1631272.1631361). 106
- A. Joly and O. Buisson. 2011. Random maximum margin hashing. In *CVPR*, pp. 873–880. DOI: [10.1109/CVPR.2011.5995709](https://doi.org/10.1109/CVPR.2011.5995709). 133
- D. Joshi, R. Datta, E. Fedorovskaya, Q.-T. Luong, J. Z. Wang, J. Li, and J. Luo. 2011. Aesthetics and emotions in images. *Signal Processing Magazine, IEEE*, 28(5): 94–115. DOI: [10.1109/MSP.2011.941851](https://doi.org/10.1109/MSP.2011.941851). 150, 156

- T. Judd, K. Ehinger, F. Durand, and A. Torralba. 2009. Learning to predict where humans look. In *ICCV*, pp. 2106–2113. DOI: [10.1109/ICCV.2009.5459462](https://doi.org/10.1109/ICCV.2009.5459462). 222, 227
- T. Kalker. 2007. A cryptographic method for secure watermark detection. In *Information Hiding*, volume 4437 of *Lecture Notes in Computer Science*, pp. 26–41. Springer. DOI: [10.1155/2007/78943](https://doi.org/10.1155/2007/78943). 81
- O. Kallenberg. 2006. *Foundations of Modern Probability*. Springer Science & Business Media. DOI: [10.1007/b98838](https://doi.org/10.1007/b98838). 185
- V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst. 2014. Matrix completion on graphs. *arXiv:1408.1717*. 58, 63, 64
- Y. Kamarianakis and P. Prastacos. 2003. Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches. *Transportation Research Record: Journal of the Transportation Research Board*, (1857): 74–84. DOI: [10.3141/1857-09](https://doi.org/10.3141/1857-09). 186
- M. Kantarcioglu, W. Jiang, Y. Liu, and B. Malin. 2008. A cryptographic approach to securely share and query genomic sequences. *IEEE Transactions on Information Technology in Biomedicine*, 12(5): 606–617. DOI: [10.1109/TTTB.2007.908465](https://doi.org/10.1109/TTTB.2007.908465). 83
- A. Kapoor, P. Shenoy, and D. Tan. 2008. Combining brain computer interfaces with vision for object categorization. In *CVPR*, pp. 1–8. DOI: [10.1109/CVPR.2008.4587618](https://doi.org/10.1109/CVPR.2008.4587618). 220
- A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *CVPR*, pp. 1725–1732. DOI: [10.1109/CVPR.2014.223](https://doi.org/10.1109/CVPR.2014.223). 10, 22
- Y. Ke, R. Sukthankar, and L. Huston. 2004. Efficient near-duplicate detection and sub-image retrieval. In *ACM Multimedia*, pp. 869–876. DOI: [10.1145/1027527.1027729](https://doi.org/10.1145/1027527.1027729). 105
- E. G. Kehoe, J. M. Toomey, J. H. Balsters, and A. L. W. Bokde. 2012. Personality modulates the effects of emotional arousal and valence on brain activation. *Social Cognitive & Affective Neuroscience*, 7: 858–870. DOI: [10.1093/scan/nsr059](https://doi.org/10.1093/scan/nsr059). 244
- A. Kendon. 1990. *Conducting Interaction: Patterns of Behavior in Focused Encounters*, volume 7. Cambridge University Press Archive, Cambridge. 51
- A. D. Keromytis. 2009. A survey of voice over IP security research. In *Information Systems Security*, volume 5905 of *Lecture Notes in Computer Science*, pp. 1–17. Springer. DOI: [10.1007/978-3-642-10772-6_1](https://doi.org/10.1007/978-3-642-10772-6_1). 98
- L. A. Khan, M. S. Baig, and A. M. Youssef. 2010. Speaker recognition from encrypted VoIP communications. *Elsevier Digital Investigation*, 7(1): 65–73. DOI: [10.1016/j.diin.2009.10.001](https://doi.org/10.1016/j.diin.2009.10.001). 98, 99, 100
- H. Kim, J. Wen, and J. D. Villasenor. 2007. Secure arithmetic coding. *IEEE Transactions on Signal Processing*, 55(5): 2263–2272. DOI: [10.1109/TSP.2007.892710](https://doi.org/10.1109/TSP.2007.892710). 86
- J. Kim and E. Andre. 2008. Emotion recognition based on physiological changes in music listening. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(12): 2067–2083. DOI: [10.1109/TPAMI.2008.26](https://doi.org/10.1109/TPAMI.2008.26). 242, 243

- S.-S. Kim, K.-I. Kim, and J. Won. 2011. Multi-view rendering approach for cloud-based gaming services. In *AFIN 2011, The Third International Conference on Advances in Future Internet*, pp. 102–107. ISBN 9781612081489. 300
- S. Kingsbury. 1987. Wisdom for the masses. *American Speech*, pp. 358–360. DOI: [10.2307/455412](https://doi.org/10.2307/455412). 162
- H. Kiya and M. Fujiyoshi. 2012. Signal and image processing in the encrypted domain. *ECIT Transactions on Computer and Information Technology*, 6(1): 11–18. 87, 88, 90
- A. Klapuri. 2003. Multiple fundamental frequency estimation by harmonicity and spectral smoothness. *IEEE Trans. Speech and Audio Processing*, 11(6): 804–816. <http://www.cs.tut.fi/sgn/arg/klap/multiplef0.pdf>. DOI: [10.1109/TSA.2003.815516](https://doi.org/10.1109/TSA.2003.815516). 37
- S. Koelstra and I. Patras. 2013. Fusion of facial expressions and EEG for implicit affective tagging. *Image and Vision Computing*, 31(2): 164–174. DOI: [10.1016/j.imavis.2012.10.002](https://doi.org/10.1016/j.imavis.2012.10.002). 247
- S. Koelstra, C. Mühl, M. Soleymani, J.-S. Lee, A. Yazdan, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras. 2012. DEAP: A database for emotion analysis using physiological signals. *IEEE Trans. Affective Computing*, 3(1): 18–31. DOI: [10.1109/T-AFFC.2011.15](https://doi.org/10.1109/T-AFFC.2011.15). 221, 240, 242, 247, 249
- C. Kofler, M. Larson, and A. Hanjalic. 2014. Intent-aware video search result optimization. *IEEE Transactions on Multimedia*, 16(5): 1421–1433. DOI: [10.1109/TMM.2014.2315777](https://doi.org/10.1109/TMM.2014.2315777). 140
- A. Kojima, T. Tamura, and K. Fukunaga. 2002. Natural language description of human activities from video images based on concept hierarchy of actions. *IJCV*, pp. 171–184. DOI: [10.1023/A:1020346032608](https://doi.org/10.1023/A:1020346032608). 16
- D. Kounades-Bastian, L. Girin, X. Alameda-Pineda, S. Gannot, and R. Horaud. 2016. A variational EM algorithm for the separation of time-varying convolutive audio mixtures. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 24(8): 1408–1423. DOI: [10.1109/TASLP.2016.2554286](https://doi.org/10.1109/TASLP.2016.2554286). 53
- D. Kounades-Bastian, L. Girin, X. Alameda-Pineda, S. Gannot, and R. Horaud. 2017. An EM algorithm for joint source separation and diarisation of multichannel convolutive mixtures. In *IEEE International Conference on Audio, Speech and Signal Processing*. New Orleans, USA. DOI: [10.1109/ICASSP.2017.7951789](https://doi.org/10.1109/ICASSP.2017.7951789). 53
- N. Krahnstoeber, M.-C. Chang, and W. Ge. 2011. Gaze and body pose estimation from a distance. In *AVSS*, pp. 11–16. DOI: [10.1109/AVSS.2011.6027285](https://doi.org/10.1109/AVSS.2011.6027285). 57
- G. Krieger, I. Rentschler, G. Hauske, K. Schill, and C. Zetzsche. 2000. Object and scene analysis by saccadic eye-movements: An investigation with higher-order statistics. *Spatial Vision*, 13(2–3): 201–214. DOI: [10.1163/156856800741216](https://doi.org/10.1163/156856800741216). 230
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *NIPS*, pp. 1097–1105. DOI: [10.1145/3065386](https://doi.org/10.1145/3065386). 3, 5, 9, 220

- H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. 2011. HMDB: A large video database for human motion recognition. In *ICCV*, pp. 2556–2563. DOI: [10.1109/ICCV.2011.6126543](https://doi.org/10.1109/ICCV.2011.6126543). 19, 21
- G. Kuhn, B. Tatler, and G. Cole. 2009. You look where I look! Effect of gaze cues on overt and covert attention in misdirection. *Visual Cognition*, 17(6–7): 925–944. DOI: [10.1080/13506280902826775](https://doi.org/10.1080/13506280902826775). 223
- B. Kulis and T. Darrell. December 2009. Learning to hash with binary reconstructive embeddings. In *NIPS*, pp. 1042–1050. 124
- B. Kulis and K. Grauman. October 2009. Kernelized locality-sensitive hashing for scalable image search. In *ICCV*, pp. 2130–2137. DOI: [10.1109/TPAMI.2011.219](https://doi.org/10.1109/TPAMI.2011.219). 133
- A. Kumar, P. Dighe, S. Chaudhuri, R. Singh, and B. Raj. 2012. Audio event detection from acoustic unit occurrence patterns. In *Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 489–492. DOI: [10.1109/ICASSP.2012.6287923](https://doi.org/10.1109/ICASSP.2012.6287923). 39
- A. Kumar, R. Singh, and B. Raj. 2014. Detecting sound objects in audio recordings. In *22nd European Signal Processing Conference, EUSIPCO 2014, Lisbon, Portugal, September 1–5, 2014*, pp. 905–909. 47
- H. Kwak, C. Lee, H. Park, and S. Moon. 2010. What is Twitter, a social network or a news media? In *WWW '10*, pp. 591–600. DOI: [10.1145/1772690.1772751](https://doi.org/10.1145/1772690.1772751). 213
- R. L. Lagendijk, Z. Erkin, and M. Barni. 2013. Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation. *IEEE Signal Processing Magazine*, 30(1): 82–105. DOI: [10.1109/MSP.2012.2219653](https://doi.org/10.1109/MSP.2012.2219653). 77, 104
- O. Lanz. 2006. Approximate Bayesian multibody tracking. *IEEE TPAMI*, 28: 1436–1449. DOI: [10.1109/TPAMI.2006.177](https://doi.org/10.1109/TPAMI.2006.177). 59, 71
- B. M. Lapham. 2014. Hawkes processes and some financial applications. Thesis, University of Cape Town. 208
- I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. 2008. Learning realistic human actions from movies. In *CVPR*, pp. 1–8. DOI: [10.1109/CVPR.2008.4587756](https://doi.org/10.1109/CVPR.2008.4587756). 21
- A. Lathey and P. K. Atrey. 2015. Image enhancement in encrypted domain over cloud. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 11(3): 38. DOI: [10.1145/2656205](https://doi.org/10.1145/2656205). 89, 90
- A. Lathey, P. Atrey, and N. Joshi. 2013. Homomorphic low pass filtering over cloud. In *IEEE International Conference on International Conference on Semantic Computing*, pp. 310–313. DOI: [10.1109/ICSC.2013.60](https://doi.org/10.1109/ICSC.2013.60). 89, 90
- P. J. Laub, T. Taimre, and P. K. Pollett. 2015. Hawkes processes. *arXiv:1507.02822*. 206
- J.-P. Laulajainen, T. Sutinen, S. Järvinen, and S. Jarvinen. Apr. 2006. Experiments with QoS-aware gaming-on-demand service. In *20th International Conference on Advanced Information Networking and Applications*, volume 1, pp. 805–810. DOI: [10.1109/AINA.2006.175](https://doi.org/10.1109/AINA.2006.175). 308

- R. Lazzeretti, J. Guajardo, and M. Barni. 2012. Privacy preserving ECG quality evaluation. In *Proceedings of the 14th ACM Workshop on Multimedia and Security*, pp. 165–174. ACM. DOI: [10.1145/2361407.2361435](https://doi.org/10.1145/2361407.2361435). 83
- Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. 2011. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR*, pp. 3361–3368. DOI: [10.1145/2361407.2361435](https://doi.org/10.1145/2361407.2361435). 14
- B. B. LeCun, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. 1990. Handwritten digit recognition with a back-propagation network. In *NIPS*, vol. 2. 5
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. 2001. Gradient-based learning applied to document recognition. In *Intelligent Signal Processing*, pp. 306–351. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791). 5
- C. M. Lee and S. S. Narayanan. 2005. Toward detecting emotions in spoken dialogs. *Speech and Audio Processing*, 13(2): 293–303. DOI: [10.1109/TSA.2004.838534](https://doi.org/10.1109/TSA.2004.838534). 221
- H. Lei, J. Choi, A. Janin, and G. Friedland. May 2011. Persona linking: Matching uploaders of videos across accounts. In *Proceedings of ICASSP*, pp. 2404–2407. 39
- H. Lei, J. Choi, and G. Friedland. 2012. Multimodal city-verification on Flickr videos using acoustic and textual features. In *Proceedings of ICASSP*, pp. 2273–2276. DOI: [10.1109/ICASSP.2012.6288367](https://doi.org/10.1109/ICASSP.2012.6288367). 39
- H. Lejsek, F. H. Asmundsson, B. P. Jónsson, and L. Amsaleg. May 2009. NV-tree: An efficient disk-based index for approximate search in very large high-dimensional collections. *IEEE Trans. PAMI*, 31(5): 869–883. DOI: [10.1109/TPAMI.2008.130](https://doi.org/10.1109/TPAMI.2008.130). 112, 114
- B. Lepri, R. Subramanian, K. Kalimeri, J. Staiano, F. Pianesi, and N. Sebe. 2012. Connecting meeting behavior with Extraversion—A systematic study. *IEEE Trans. Affective Computing*, 3(4): 443–455. DOI: [10.1109/T-AFFC.2012.17](https://doi.org/10.1109/T-AFFC.2012.17). 238, 240
- J. Leskovec, A. Singh, and J. Kleinberg. 2006. Patterns of influence in a recommendation network. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 380–389. Springer. DOI: [10.1007/11731139_44](https://doi.org/10.1007/11731139_44). 152
- H. J. Levesque, R. Reiter, Y. Lespérance, F. Lin, and R. B. Scherl. 1997. GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming*, P1(1–3): 59–83. DOI: [10.1016/S0743-1066\(96\)00121-5](https://doi.org/10.1016/S0743-1066(96)00121-5). 165
- M. S. Lew, N. Sebe, C. Djeraba, and R. Jain. 2006. Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2(1): 1–19. DOI: [10.1145/1126004.1126005](https://doi.org/10.1145/1126004.1126005). 85
- J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou. 2010. Fuzzy keyword search over encrypted data in cloud computing. In *Proceedings of IEEE INFOCOM*, pp. 441–445. DOI: [10.1007/978-3-642-38562-9_74](https://doi.org/10.1007/978-3-642-38562-9_74). 260
- P. Li, X. Yu, Y. Liu, and T. Zhang. 2014. Crowdsourcing fraud detection algorithm based on Ebbinghaus forgetting curve. *International Journal of Security and Its Applications*, 8(1): 283–290. DOI: [10.14257/ijasia.2014.8.1.26](https://doi.org/10.14257/ijasia.2014.8.1.26). 262

- Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo. 2016a. Action recognition by learning deep multi-granular spatio-temporal video representation. In *ICMR*, 159–166. DOI: [10.1145/2911996.2912001](https://doi.org/10.1145/2911996.2912001). 12
- W. Li, Y. Zhang, Y. Sun, W. Wang, W. Zhang, and X. Lin. 2016b. Approximate nearest neighbor search on high dimensional data—Experiments, analyses, and improvement (v1.0). *arXiv:1610.02455*. 134
- X. Li, C. G. Snoek, and M. Worring. 2009. Learning social tag relevance by neighbor voting. *IEEE Transactions on Multimedia*, 11(7): 1310–1322. DOI: [10.1109/TMM.2009.2030598](https://doi.org/10.1109/TMM.2009.2030598). 148
- Z. Li, E. Gavves, M. Jain, and C. G. Snoek. 2016c. VideoLSTM convolves, attends and flows for action recognition. *arXiv:1607.01794*. 13
- C. Liao, T. Hou, T. Lin, Y. Cheng, C. H. A. Erbad, and N. Venkatasubramania. 2014. SAIS: Smartphone augmented infrastructure sensing for public safety and sustainability in smart cities. In *Proceedings of ACM International Workshop on Emerging Multimedia Applications and Services for Smart Cities (EMASC)*, pp. 3–8. DOI: [10.1145/2661704.2661706](https://doi.org/10.1145/2661704.2661706). 268, 269, 270, 271, 272
- L. Lin, X. Liao, G. Tan, H. Jin, X. Yang, W. Zhang, and B. Li. 2014. LiveRender: A Cloud Gaming System Based on Compressed Graphics Streaming. In *Proceedings of the ACM International Conference on Multimedia*, MM '14, pp. 347–356. ACM, New York. DOI: [10.1145/2647868.2654943](https://doi.org/10.1145/2647868.2654943). 307
- T. Lin, T. Lin, C. Hsu, and C. King. 2013. Context-aware decision engine for mobile cloud offloading. In *Proceedings of IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 111–116. DOI: [10.1109/WCNCW.2013.6533324](https://doi.org/10.1109/WCNCW.2013.6533324). 258
- F. Lingensfelder, J. Wagner, E. André, G. McKeown, and W. Curran. 2014. An event driven fusion approach for enjoyment recognition in real-time. In *ACMMM*, pp. 377–386. DOI: [10.1145/2647868.2654924](https://doi.org/10.1145/2647868.2654924). 52
- M. Lipczak, M. Trevisiol, and A. Jaimes. 2013. Analyzing favorite behavior in Flickr. In *International Conference on Multimedia Modeling*, pp. 535–545. Springer. DOI: [10.1007/978-3-642-35725-1_49](https://doi.org/10.1007/978-3-642-35725-1_49). 148
- S. Liu, P. Cui, H. Luan, W. Zhu, S. Yang, and Q. Tian. 2013a. Social visual image ranking for web image search. In *International Conference on Multimedia Modeling*, pp. 239–249. Springer. DOI: [10.1007/978-3-642-35725-1_22](https://doi.org/10.1007/978-3-642-35725-1_22). 149
- S. Liu, P. Cui, W. Zhu, S. Yang, and Q. Tian. 2014. Social embedding image distance learning. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 617–626. ACM. DOI: [10.1145/2647868.2654905](https://doi.org/10.1145/2647868.2654905). 148, 149
- S. Liu, P. Cui, W. Zhu, and S. Yang. 2015. Learning socially embedded visual representation from scratch. In *Proceedings of the 23rd ACM International conference on Multimedia*, pp. 109–118. ACM. DOI: [10.1145/2733373.2806247](https://doi.org/10.1145/2733373.2806247). 149
- X. Liu, J. He, and B. Lang. July 2013b. Reciprocal hash tables for nearest neighbor search. In *Proceedings of the 27th Association for the Advancement of Artificial Intelligence (AAAI) Conference on Artificial Intelligence*, pp. 626–632. 118

- Y. Liu and Z. Shi. 2016. Boosting video description generation by explicitly translating from frame-level captions. In *ACM Multimedia*, pp. 631–634. DOI: [10.1145/2964284.2967298](https://doi.org/10.1145/2964284.2967298). 17, 28
- Y. Liu, Y. Guo, and C. Liang. 2008. A survey on peer-to-peer video streaming systems. *Peer-to-peer Networking and Applications*, 1(1): 18–29. DOI: [10.1007/s12083-007-0006-y](https://doi.org/10.1007/s12083-007-0006-y). 259
- B. Logan et al. 2000. Mel frequency cepstral coefficients for music modeling. In *International Symposium on Music Information Retrieval*, volume 28, p. 5. 37
- LogMeIn. July 2012. LogMeIn web page. <http://secure.logmein.com/>. 291
- J. Long, E. Shelhamer, and T. Darrell. 2015. Fully convolutional networks for semantic segmentation. In *CVPR*, p. 3431–3440. DOI: [10.1109/TPAMI.2016.2572683](https://doi.org/10.1109/TPAMI.2016.2572683). 3, 9
- A. Loui, J. Luo, S.-F. Chang, D. Ellis, W. Jiang, L. Kennedy, K. Lee, and A. Yanagawa. 2007. Kodak’s consumer video benchmark data set: Concept definition and annotation. In *ACM Multimedia Information Retrieval (MIR) Workshop*, pp. 245–254. DOI: [10.1145/1290082.1290117](https://doi.org/10.1145/1290082.1290117). 20
- D. Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2): 91–110. 106, 107, 275
- W. Lu, A. Swaminathan, A. L. Varna, and M. Wu. 2009a. Enabling search over encrypted multimedia databases. *Proceedings of International Society for Optics and Photonics, SPIE, Media Forensics and Security*, pp. 7254–7318. DOI: [10.1117/12.806980](https://doi.org/10.1117/12.806980). 82, 85, 86, 90
- W. Lu, A. L. Varna, A. Swaminathan, and M. Wu. 2009b. Secure image retrieval through feature protection. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1533–1536. DOI: [10.1109/ICASSP.2009.4959888](https://doi.org/10.1109/ICASSP.2009.4959888). 85, 86, 90
- W. Lu, A. L. Varna, and M. Wu. 2010. Security analysis for privacy preserving search of multimedia. In *IEEE International Conference on Image Processing*, pp. 2093–2096. DOI: [10.1109/ICIP.2010.5653399](https://doi.org/10.1109/ICIP.2010.5653399). 86
- W. Lu, A. Varna, and M. Wu. 2011. Secure video processing: Problems and challenges. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5856–5859. DOI: [10.1109/ICASSP.2011.5947693](https://doi.org/10.1109/ICASSP.2011.5947693). 77
- P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. 2010. The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *CVPR Workshops*, pp. 94–101. DOI: [10.1109/CVPRW.2010.5543262](https://doi.org/10.1109/CVPRW.2010.5543262). 221
- Y. Luo, S. Ye, and S. Cheung. 2010. Anonymous subject identification in privacy-aware video surveillance. In *Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 83–88. DOI: [10.1109/ICME.2010.5583561](https://doi.org/10.1109/ICME.2010.5583561). 92
- Q. Lv, M. Charikar, and K. Li. Nov. 2004. Image similarity search with compact data structures. In *CIKM*, pp. 208–217. DOI: [10.1145/1031171.1031213](https://doi.org/10.1145/1031171.1031213). 120

- Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li. 2007. Multi-probe LSH: Efficient indexing for high-dimensional similarity search. In *Proceedings of the International Conference on Very Large DataBases*, pp. 950–961. 118
- M. S. Magnusson. 2000. Discovering hidden time patterns in behavior: T-patterns and their detection. *Behavior Research Methods, Instruments, & Computers*, 32(1): 93–110. DOI: [10.3758/BF03200792](https://doi.org/10.3758/BF03200792). 187
- R. Maher and J. Beauchamp. 1994. Fundamental frequency estimation of musical signals using a two-way mismatch procedure. *The Journal of the Acoustical Society of America*, 95(4): 2254–2263. DOI: [10.1121/1.408685](https://doi.org/10.1121/1.408685). 37
- Y. Malkov, A. Ponomarenko, A. Logvinov, and V. Krylov. 2014. Approximate nearest neighbor algorithm based on navigable small world graphs. *Information Systems*, 45: 61–68. DOI: [10.1016/j.is.2013.10.006](https://doi.org/10.1016/j.is.2013.10.006). 133
- Y. A. Malkov and D. A. Yashunin. March 2016. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *arXiv:1603.09320*. 133
- M. Mandel and D. P. W. Ellis. Sept. 2008. Multiple-instance learning for music information retrieval. In *Proceedings ISMIR*, pp. 577–582. Philadelphia, PA. <http://www.ee.columbia.edu/~dpwe/pubs/MandelE08-MImusic.pdf>. 38
- M. I. Mandel and D. P. W. Ellis. Sept. 2005. Song-level features and support vector machines for music classification. In *Proceedings International Conference on Music Information Retrieval ISMIR*, pp. 594–599. London. <http://www.ee.columbia.edu/~dpwe/pubs/ismir05-svm.pdf>. 38
- C. D. Manning, P. Raghavan, and H. Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press. 116
- Y. Mao and M. Wu. 2006. A joint signal processing and cryptographic approach to multimedia encryption. *Proceedings of IEEE Transactions on Image Processing*, 15(7): 2061–2075. DOI: [10.1109/TIP.2006.873426](https://doi.org/10.1109/TIP.2006.873426). 92, 93, 96
- O. Maron and T. Lozano-Perez. 1998. A framework for multiple-instance learning. In *Advances in Neural Information Processing Systems*, pp. 570–576. 48
- M. Marszalek, I. Laptev, and C. Schmid. 2009. Actions in context. In *CVPR*. pp. 2929–2936. DOI: [10.1109/CVPRW.2009.5206557](https://doi.org/10.1109/CVPRW.2009.5206557). 21
- J. Martinez, H. H. Hoos, and J. J. Little. 2014. Stacked quantizers for compositional vector compression. *arXiv:1411.2173*. 131
- I. Martínez-Ponte, X. Desurmont, J. Meessen, and J.-F. Delaigle. 2005. Robust human face hiding ensuring privacy. In *Proceedings of International Workshop on Image Analysis for Multimedia Interactive Services*, p. 4. DOI: [10.1.1.72.4758](https://doi.org/10.1.1.72.4758). 91
- B. Matei, Y. Shan, H. Sawhney, Y. Tan, R. Kumar, D. Huber, and M. Hebert. July 2006. Rapid object indexing using locality sensitive hashing and joint 3D-signature space estimation. *IEEE Trans. PAMI*, 28(7): 1111–1126. DOI: [10.1109/TPAMI.2006.148](https://doi.org/10.1109/TPAMI.2006.148). 105

- A. Matic, V. Osmani, A. Maxhuni, and O. Mayora. 2012. Multi-modal mobile sensing of social interactions. In *PervasiveHealth*. (Pervasive Computing Technologies for Healthcare), IEEE, May 2012. DOI: [10.4108/icst.pervasivehealth.2012.248689](https://doi.org/10.4108/icst.pervasivehealth.2012.248689). 55
- A. Matviienko, A. Löcken, A. El Ali, W. Heuten, and S. Boll. 2016. NaviLight: Investigating ambient light displays for turn-by-turn navigation in cars. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 283–294. ACM. DOI: [10.1145/2935334.2935359](https://doi.org/10.1145/2935334.2935359). 188
- J. McCarthy, and P. Hayes. 1968. *Some philosophical problems from the standpoint of artificial intelligence*. Stanford University: Stanford Artificial Intelligence Project. 165, 167
- W. McCulloch and W. Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5(4): 115–133. DOI: [10.1007/BF02478259](https://doi.org/10.1007/BF02478259). 5
- MediaEval. 2010. MediaEval’s video CLEF-Placing task. <http://www.multimediaeval.org/placing/placing.html>. 39
- D. Meilander, F. Glinka, S. Gorlatch, L. Lin, W. Zhang, X. Liao, and D. Meil. Apr. 2014. Bringing mobile online games to clouds. In *2014 IEEE Conference on Computer Communications Workshops*, pp. 340–345. DOI: [10.1109/INFCOMW.2014.6849255](https://doi.org/10.1109/INFCOMW.2014.6849255). 308
- D. K. Mellinger. 1991. *Event formation and separation in musical sound*. PhD thesis, CCRMA, Stanford University. 37
- Merriam-Webster. 2003. *Merriam-Webster’s Collegiate Dictionary*. Merriam-Webster. 166, 167
- R. Mertens, P. S. Huang, L. Gottlieb, G. Friedland, and A. Divakaran. 2011a. On the applicability of speaker diarization to audio concept detection for multimedia retrieval. In *Proceedings of the IEEE International Symposium on Multimedia*, pp. 446–451. DOI: [10.1109/ISM.2011.79](https://doi.org/10.1109/ISM.2011.79). 43
- R. Mertens, H. Lei, L. Gottlieb, G. Friedland, and A. Divakaran. 2011b. Acoustic super models for large scale video event detection. In *Proceedings of the Joint ACM Workshop on Modeling and Representing Events*, pp. 19–24. DOI: [10.1145/2072508.2072513](https://doi.org/10.1145/2072508.2072513). 36
- I. Mervielde, F. De Fruyt, and S. Jarmuz. 1998. Linking openness and intellect in childhood and adulthood. In *Parental Descriptions of Child Personality: Developmental Antecedents of the Big Five*, pp. 105–126. 246
- P. Messaris. 1996. *Visual Persuasion: The Role of Images in Advertising*. Sage Publications. 187
- P. B. Miltersen, N. Nisan, S. Safra, and A. Wigderson. August 1998. On data structures and asymmetric communication complexity. *Journal of Computer and System Sciences*, 57: 37–49. DOI: [10.1006/jcss.1998.1577](https://doi.org/10.1006/jcss.1998.1577). 108
- D. Mishra, M. E. Zarki, A. Erbad, C.-H. Hsu, N. Venkatasubramanian, and M. El Zarki. May 2014. Clouds + games: A multifaceted approach. *Internet Computing, IEEE*, 18(3): 20–27. DOI: [10.1109/MIC.2014.20](https://doi.org/10.1109/MIC.2014.20). 289, 290, 291
- M. Mishra, A. Das, P. Kulkarni, and A. Sahoo. 2012. Dynamic resource management using virtual machine migrations. *IEEE Communications Magazine*, 50(9): 34–40. DOI: [10.1109/MCOM.2012.6295709](https://doi.org/10.1109/MCOM.2012.6295709). 259

- S. Mishra, M.-A. Rizoïu, and L. Xie. 2016. Feature driven and point process approaches for popularity prediction. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management*, pp. 1069–1078. DOI: [10.1145/2983323.2983812](https://doi.org/10.1145/2983323.2983812). 209, 211, 213
- M. Mohanty, P. K. Atrey, and W.-T. Tsang. 2012. Secure cloud-based medical data visualization over cloud. In *Proceedings of ACM International Conference on Multimedia*, pp. 1105–1108. DOI: [10.1145/2393347.2396394](https://doi.org/10.1145/2393347.2396394). 89, 90
- M. Mohanty, W.-T. Tsang, and P. K. Atrey. 2013. Scale me, crop me, know me not: Supporting scaling and cropping in secret image sharing. In *IEEE International Conference on Multimedia and Expo*, pp. 1–6. DOI: [10.1109/ICME.2013.6607567](https://doi.org/10.1109/ICME.2013.6607567). 89, 90
- D. Moise, D. Shestakov, G. P. Gudmundsson, and L. Amsaleg. April 2013. Indexing and searching 100M images with Map-Reduce. In *ICMR*, pp. 17–24. DOI: [10.1145/2461466.2461470](https://doi.org/10.1145/2461466.2461470). 113
- J. Møller and J. G. Rasmussen. 2005. Perfect simulation of Hawkes processes. *Advances in Applied Probability*, 37(3): 629–646. DOI: [10.1017/S0001867800000392](https://doi.org/10.1017/S0001867800000392). 208
- M. Montanari, S. Mehrotra, and N. Venkatasubramanian. 2007. Architecture for an automatic customized warning system. In *2007 IEEE Intelligence and Security Informatics*, pp. 32–39. IEEE. DOI: [10.1109/ISI.2007.379530](https://doi.org/10.1109/ISI.2007.379530). 177, 182
- J. A. Moorer. 1975. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer*. PhD thesis, Department of Music, Stanford University. 37
- N. Moray and T. Sheridan. 2004. Oil sont les neiges d’antan? In *Human Performance, Situation Awareness and Automation II*. D. A. Vincenzi, H. Mouloua, and P. Hancock, eds. Lawrence Erlbaum Associates, Marwah NJ. 165, 167
- C. Moreno, N. Tizon, and M. Preda. 2012. Mobile cloud convergence in GaaS: A business model proposition. In *2012 45th Hawaii International Conference on System Science (HICSS)*, pp. 1344–1352. DOI: [10.1109/HICSS.2012.433](https://doi.org/10.1109/HICSS.2012.433). 290
- C. P. Moreno, A. G. Antolín, and F. D. Fernando. 2001. Recognizing voice over IP: A robust front-end for speech recognition on the world wide web. *Proceedings of IEEE Transactions on Multimedia*, 3(2): 209–218. DOI: [10.1109/6046.923820](https://doi.org/10.1109/6046.923820). 97
- P. J. Moreno, B. Raj, and R. M. Stern. 1996. A vector taylor series approach for environment-independent speech recognition. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pp. 733–736. IEEE. DOI: [10.1109/ICASSP.1996.543225](https://doi.org/10.1109/ICASSP.1996.543225). 36
- F. Morstatter, J. Pfeffer, H. Liu, and K. M. Carley. 2013. Is the sample good enough? Comparing data from Twitter’s streaming API with Twitter’s firehose. http://scholar.google.de/scholar.bib?q=info:NkS2af1rqyQJ:scholar.google.com/&output=citation&hl=de&as_sdt=0,5&ct=citation&cd=0. 144
- M. Muja and D. G. Lowe. February 2009. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP*, pp. 331–340. DOI: [10.5220/0001787803310340](https://doi.org/10.5220/0001787803310340). 116, 117

- M. Muja and D. G. Lowe. 2014. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Trans. PAMI*, p. 36. DOI: [10.1109/TPAMI.2014.2321376](https://doi.org/10.1109/TPAMI.2014.2321376). 117
- M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard. Oct. 2011. Signal processing for music analysis. *IEEE Journal of Selected Topics in Signal Processing*, 5(6): 1088–1110. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5709966. DOI: [10.1109/JSTSP.2011.2112333](https://doi.org/10.1109/JSTSP.2011.2112333). 38
- E. Murphy-Chutorian and M. M. Trivedi. 2009. Head pose estimation in computer vision: A survey. *IEEE Trans. PAMI*, 31(4): 607–626. DOI: [10.1109/TPAMI.2008.106](https://doi.org/10.1109/TPAMI.2008.106). 219
- B. Myers, S. Hudson, and R. Pausch. 2000. Past, present, and future of user interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1): 3–28. DOI: [10.1145/344949.344959](https://doi.org/10.1145/344949.344959). 170
- G. Mysore, P. Smaragdis, and B. Raj. 2010. Non-negative hidden-Markov modeling of audio with application to source separation. In *Proceedings of 9th International Conference on Latent Variable Analysis and Source Separation (LVA/ICA)*, pp. 140–148. 37
- M. Naaman. 2012. Social multimedia: Highlighting opportunities for search and mining of multimedia data in social media applications. *Multimedia Tools and Applications*, 56(1): 9–34. DOI: [10.1007/s11042-010-0538-7](https://doi.org/10.1007/s11042-010-0538-7). 145
- X. Naturel and P. Gros. June 2008. Detecting repeats for video structuring. *Multimedia Tools and Applications*, 38: 233–252. DOI: [10.1007/s11042-007-0180-1](https://doi.org/10.1007/s11042-007-0180-1). 120
- A. Nazari Shirehjini. 2006. Situation modelling: A domain analysis and user study. In *2nd IET International Conference on Intelligent Environments, 2006*, volume 2, pp. 193–199. IET. DOI: [10.1049/cp:20060695](https://doi.org/10.1049/cp:20060695). 165
- R.-A. Negoescu and D. Gatica-Perez. 2010. Modeling Flickr communities through probabilistic topic-based analysis. *IEEE Transactions on Multimedia*, 12(5): 399–416. DOI: [10.1109/TMM.2010.2050649](https://doi.org/10.1109/TMM.2010.2050649). 148
- E. M. Newton, L. Sweeney, and B. Malin. 2005. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17(2): 232–243. DOI: [10.1109/TKDE.2005.32](https://doi.org/10.1109/TKDE.2005.32). 92, 93
- B. Neyshabur and N. Srebro. 2015. On symmetric and asymmetric LSHs for inner product search. In *ICML*, pp. 1926–1934. 133
- J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *CVPR*, pp. 4694–4702. DOI: [10.1109/CVPR.2015.7299101](https://doi.org/10.1109/CVPR.2015.7299101). 12, 24
- W. Ng. 2009. Clarifying the relation between neuroticism and positive emotions. *Personality and Individual Differences*, 47(1): 69–72. DOI: [10.1016/j.paid.2009.01.049](https://doi.org/10.1016/j.paid.2009.01.049). 244
- J. C. Niebles, H. Wang, and L. Fei-Fei. 2008. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3): 299–318. DOI: [10.1007/s11263-007-0122-4](https://doi.org/10.1007/s11263-007-0122-4). 47

- J. C. Niebles, C.-W. Chen, and L. Fei-Fei. 2010. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, pp. 392–405. DOI: [10.1007/978-3-642-15552-9_29](https://doi.org/10.1007/978-3-642-15552-9_29). 21
- D. Nistér and H. Stewénus. June 2006. Scalable recognition with a vocabulary tree. In *CVPR*, pp. 2161–2168. DOI: [10.1109/CVPR.2006.264](https://doi.org/10.1109/CVPR.2006.264). 116
- M. Norouzi and D. Fleet. June 2013. Cartesian k-means. In *CVPR*, pp. 3017–3024. DOI: [10.1109/CVPR.2013.388](https://doi.org/10.1109/CVPR.2013.388). 131
- C. Norris, J. Moran, and G. Armstrong (eds.). 1998. CCTV: A new battleground for privacy. *Surveillance, closed-circuit television and social control*, pp. 243–254. 91
- M. Nouh, A. Almaatouq, A. Alabdulkareem, V. K. Singh, E. Shmueli, M. Alsaleh, A. Alarifi, A. Alfaris, and A. S. Pentland. 2014. Social information leakage: Effects of awareness and peer pressure on user behavior. In *Human Aspects of Information Security, Privacy, and Trust*, pp. 352–360. Springer. DOI: [10.1007/978-3-319-07620-1_31](https://doi.org/10.1007/978-3-319-07620-1_31). 188
- E. Nowak, F. Jurie, and B. Triggs. 2006. Sampling strategies for bag-of-features image classification. *Computer Vision—ECCV 2006*, pp. 490–503. DOI: [10.1007/11744085_38](https://doi.org/10.1007/11744085_38). 167
- Y. Ogata. 1981. On Lewis' simulation method for point processes. *IEEE Transactions on Information Theory*, 27(1): 23–31. DOI: [10.1109/TIT.1981.1056305](https://doi.org/10.1109/TIT.1981.1056305). 202, 204, 209
- Y. Ogata. 1988. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401). DOI: [10.1080/01621459.1988.10478560](https://doi.org/10.1080/01621459.1988.10478560). 200, 209
- Y. Ogata, R. S. Matsuúra, and K. Katsura. 1993. Fast likelihood computation of epidemic type aftershock-sequence model. *Geophysical Research Letters*, 20(19): 2143–2146. DOI: [10.1029/93GL02142](https://doi.org/10.1029/93GL02142). 209
- A. Ojala and P. Tyrvaínen. Jul. 2011. Developing cloud business models: A case study on cloud gaming. *IEEE Software*, 28(4): 42–47. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5741005>. DOI: [10.1109/MS.2011.51](https://doi.org/10.1109/MS.2011.51). 290
- A. Oliva and A. Torralba. 2001. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3): 145–175. DOI: [10.1023/A:1011139631724](https://doi.org/10.1023/A:1011139631724). 132
- D. Omercevic, O. Drbohlav, and A. Leonardis. October 2007. High-dimensional feature matching: employing the concept of meaningful nearest neighbors. In *ICCV*, pp. 1–8. DOI: [10.1109/ICCV.2007.4408880](https://doi.org/10.1109/ICCV.2007.4408880). 106
- OnLive. January 2015. OnLive web page. <http://www.onlive.com/>. 287, 291
- M. Osadchy, B. Pinkas, A. Jarrous, and B. Moskovich. 2010. SCiFI—A system for secure face identification. In *Proceedings of IEEE Symposium on Security and Privacy*, pp. 239–254. DOI: [10.1109/SP.2010.39](https://doi.org/10.1109/SP.2010.39). 94, 96
- P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quéenot. 2014. TRECVID 2014—An overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2014*, National Institute of Standards and Technology (NIST). 19, 21, 23

- P. Over, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, G. Quénot, and R. Ordelman. 2015. TRECVID 2015—An overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2015*, NIST. 23
- T. Ozaki. 1979. Maximum likelihood estimation of Hawkes' self-exciting point processes. *Annals of the Institute of Statistical Mathematics*, 31(1): 145–155. DOI: [10.1007/BF02480272](https://doi.org/10.1007/BF02480272). 198
- P. Paillier. 1999. Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques*, volume 1592 of *Lecture Notes in Computer Science*, pp. 223–238. Springer. DOI: [10.1007/3-540-48910-X_16](https://doi.org/10.1007/3-540-48910-X_16). 79, 87
- Y. Pan, T. Yao, T. Mei, H. Li, C.-W. Ngo, and Y. Rui. 2014. Click-through-based cross-view learning for image search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 717–726. ACM. DOI: [10.1145/2600428.2609568](https://doi.org/10.1145/2600428.2609568). 17, 148
- Y. Pan, Y. Li, T. Yao, T. Mei, H. Li, and Y. Rui. 2016a. Learning deep intrinsic video representation by exploring temporal coherence and graph structure. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3832–3838. 14
- Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. 2016b. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, pp. 4594–4602. 15, 16, 17, 28
- Y. Pan, T. Yao, H. Li, and T. Mei. 2016c. Video captioning with transferred semantic attributes. *arXiv:1611.07675*. 17
- S. Pancoast and M. Akbacak. 2011. Bag-of-audio-words approach for multimedia event classification. In *Proceedings of Interspeech*, pp. 2105–2108. 46
- P. F. Panter and W. Dite. Jan. 1951. Quantizing distortion in pulse-count modulation with nonuniform spacing of levels. *Proceedings IRE*, 39: 44–48. DOI: [10.1109/JRPROC.1951.230419](https://doi.org/10.1109/JRPROC.1951.230419). 117
- M. Pantic, N. Sebe, J. F. Cohn, and T. Huang. 2005. Affective multimodal human-computer interaction. In *ACMMM*, pp. 669–676. 52
- D. P. Papadopoulos, A. D. F. Clarke, F. Keller, and V. Ferrari. 2014. Training object class detectors from eye tracking data. In *ECCV*, pp. 361–376. DOI: [10.1007/978-3-319-10602-1_24](https://doi.org/10.1007/978-3-319-10602-1_24). 220, 226
- K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *ACL*, pp. 311–318. DOI: [10.3115/1073083.1073135](https://doi.org/10.3115/1073083.1073135). 27
- O. M. Parkhi, A. Vedaldi, C. V. Jawahar, and A. Zisserman. 2011. The truth about cats and dogs. *2011 International Conference on Computer Vision*, pp. 1427–1434. DOI: [10.1109/ICCV.2011.6126398](https://doi.org/10.1109/ICCV.2011.6126398). 227
- D. Parkhurst and E. Niebur. 2003. Scene content selected by active vision. *Spatial Vision*, 16(2): 125–54. DOI: [10.1163/15685680360511645](https://doi.org/10.1163/15685680360511645). 230

- P. Patel, A. Ranabahu, and A. Sheth. 2009. Service level agreement in cloud computing. In *Proceedings of International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, 1(1–10). 260
- M. A. Pathak and B. Raj. March 2012. Privacy-preserving speaker verification as password matching. In *ICASSP*, pp. 1849–1852. 105
- M. Pathak and B. Raj. 2013. Privacy-preserving speaker verification and identification using gaussian mixture models. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(2): 397–406. DOI: [10.1109/TASL.2012.2215602](https://doi.org/10.1109/TASL.2012.2215602). 99, 100
- L. Paulevé, H. Jégou, and L. Amsaleg. Aug. 2010. Locality sensitive hashing: A comparison of hash function types and querying mechanisms. *Pattern Recognition Letters*, 31(11): 1348–1358. DOI: [10.1016/j.patrec.2010.04.004](https://doi.org/10.1016/j.patrec.2010.04.004). 110, 116, 118, 119, 132
- E. M. M. Peck, B. F. Yuksel, A. Ottley, R. J. Jacob, and R. Chang. 2013. Using fNIRS brain sensing to evaluate information visualization interfaces. In *ACM Conference on Human Factors in Computing Systems*, pp. 473–482. DOI: [10.1016/j.patrec.2010.04.004](https://doi.org/10.1016/j.patrec.2010.04.004). 251
- M. Perugini and L. Di Blas. 2002. Analyzing personality-related adjectives from an eticemic perspective: The big five marker scale (BFMS) and the Italian AB5C taxonomy. *Big Five Assessment*, pp. 281–304. 222, 238, 240
- R. J. Peters, A. Iyer, L. Itti, and C. Koch. 2005. Components of bottom-up gaze allocation in natural images. *Vision Research*, 45(8): 2397–2416. DOI: [10.1016/j.visres.2005.03.019](https://doi.org/10.1016/j.visres.2005.03.019). 231
- S. Petridis, B. Martinez, and M. Pantic. 2013. The MAHNOB laughter database. *Image and Vision Computing*, 31(2): 186–202. DOI: [10.1016/j.imavis.2012.08.014](https://doi.org/10.1016/j.imavis.2012.08.014). 55
- T. T. Pham, S. Hamid Rezatofighi, I. Reid, and T.-J. Chin. 2016. Efficient point process inference for large-scale object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2837–2845. DOI: [10.1109/CVPR.2016.310](https://doi.org/10.1109/CVPR.2016.310). 193
- J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. June 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, pp. 1–8. DOI: [10.1109/CVPR.2008.4587635](https://doi.org/10.1109/CVPR.2008.4587635). 116, 119
- A. Pikrakis, T. Giannakopoulos, and S. Theodoridis. 2008. Gunshot detection in audio streams from movies by means of dynamic programming and Bayesian networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 21–24. IEEE. DOI: [10.1109/ICASSP.2008.4517536](https://doi.org/10.1109/ICASSP.2008.4517536). 38
- J. K. Pillai, V. M. Patel, R. Chellappa, and N. K. Ratha. 2011. Secure and robust iris recognition using random projections and sparse representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9): 1877–1893. 82
- A. Piva and S. Katzenbeisser. 2008. Special issue on signal processing in the encrypted domain. *Hindawi Publishing Corporation, EURASIP Journal on Information Security (eds.)*, 2007. DOI: [10.1155/2007/82790](https://doi.org/10.1155/2007/82790). 77

- G. Poliner and D. P. W. Ellis. 2007. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007(2007): 9 pages. <http://www.ee.columbia.edu/~dpwe/pubs/PoliE06-piano.pdf>. DOI: [10.1155/2007/48317](https://doi.org/10.1155/2007/48317). Special Issue on Music Signal Processing. 38
- S. Pongpaichet, V. K. Singh, R. Jain, and A. P. Pentland. 2013. Situation fencing: making geo-fencing personal and dynamic. In *2013 ACM International Workshop on Personal Data Meets Distributed Multimedia*. Association for Computing Machinery, pp. 3–10. DOI: [10.1145/2509352.2509401](https://doi.org/10.1145/2509352.2509401). 161, 177
- R. Poppe. 2010. A survey on vision-based human action recognition. *Image and Vision Computing*, pp. 976–990. DOI: [10.1016/j.imavis.2009.11.014](https://doi.org/10.1016/j.imavis.2009.11.014). 4
- D. A. Pospelov. 1986. *Situational Control: Theory and Practice* (in Russian). Nauka. 165
- I. Potamitis, S. Ntalampiras, O. Jahn, and K. Riede. 2014. Automatic bird sound detection in long real-field recordings: Applications and tools. *Applied Acoustics*, 80: 1–9. DOI: [10.1016/j.apacoust.2014.01.001](https://doi.org/10.1016/j.apacoust.2014.01.001). 38
- J. Pouwelse, P. Garbacki, D. Epema, and H. Sips. 2005. The Bittorrent P2P file-sharing system: Measurements and analysis. In *Proceedings of International Workshop on Peer-to-Peer Systems (IPTPS)*, pp. 205–216. DOI: [10.1007/11558989_19](https://doi.org/10.1007/11558989_19). 259
- J. Prins, Z. Erkin, and R. L. Lagendijk. 2006. Literature study: Signal processing in the encrypted domain. Technical report, Information and Communication Theory Group, Delft University of Technology. 77
- W. Puech, Z. Erkin, M. Barni, S. Rane, and R. L. Lagendijk. 2012. Emerging cryptographic challenges in image and video processing. *Proceedings of 19th IEEE International Conference on Image Processing*, pp. 2629–2632. DOI: [10.1109/ICIP.2012.6467438](https://doi.org/10.1109/ICIP.2012.6467438). 77, 82, 103
- Z. Qin, J. Yan, K. Ren, C. W. Chen, and C. Wang. 2014. Towards efficient privacy-preserving image feature extraction in cloud computing. In *Proceedings of the ACM International Conference on Multimedia*, pp. 497–506. ACM. DOI: [10.1145/2647868.2654941](https://doi.org/10.1145/2647868.2654941). 87, 90
- F. Qiu and J. Cho. 2006. Automatic identification of user interest for personalized search. In *Proceedings of the 15th International Conference on World Wide Web*, pp. 727–736. ACM. DOI: [10.1145/1135777.1135883](https://doi.org/10.1145/1135777.1135883). 147
- Z. Qiu, T. Yao, and T. Mei. 2016. Deep quantization: Encoding convolutional activations with deep generative model. *arXiv:1611.09502*. 9
- J. Rabin, J. Delon, and Y. Gousseau. Dec. 2008. A contrario matching of SIFT-like descriptors. In *ICPR*, pp. 1–4. DOI: [10.1109/ICPR.2008.4761371](https://doi.org/10.1109/ICPR.2008.4761371). 106
- J. Rabin, J. Delon, and Y. Gousseau. Sep. 2009. A statistical approach to the matching of local features. *SIAM Journal on Imaging Sciences*, 2(3): 931–958. DOI: [10.1137/090751359](https://doi.org/10.1137/090751359). 106
- M. Radovanović, A. Nanopoulos, and M. Ivanović. Dec. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11: 2487–2531. 106

- M. Raginsky and S. Lazebnik. 2010. Locality-sensitive binary codes from shift-invariant kernels. In *NIPS*, pp. 1509–1517. [122](#)
- A. Rahimi and B. Recht. 2007. Random features for large-scale kernel machines. In *NIPS*, pp. 1177–1184. [122](#)
- B. Raj and R. M. Stern. 2005. Missing-feature approaches in speech recognition. *Signal Processing Magazine, IEEE*, 22(5): 101–116. DOI: [10.1109/MSP.2005.1511828](#). [36](#)
- B. Raj, T. Virtanen, S. Chaudhuri, and R. Singh. 2010a. Non-negative matrix factorization based compensation of music for automatic speech recognition. In *Proceedings of Interspeech*, pp. 717–720. [37](#)
- B. Raj, T. Virtanen, S. Chaudhuri, and R. Singh. 2010b. Ungrounded non-negative independent factor analysis. In *Proceedings of Interspeech*, pp. 330–333. [37](#)
- B. Raj, R. Singh, and T. Virtanen. 2011. Phoneme-dependent NMF for speech enhancement in monaural mixtures. In *Proceedings of Interspeech*, pp. 1217–1220. [37](#)
- A. K. Rajagopal, R. Subramanian, R. L. Vieri, E. Ricci, O. Lanz, K. Ramakrishnan, and N. Sebe. 2012. An adaptation framework for head-pose classification in dynamic multi-view scenarios. In *Asian Conference on Computer Vision (ACCV)*, pp. 652–666. Springer Berlin Heidelberg. DOI: [10.1007/978-3-642-37444-9_51](#). [57](#)
- A. K. Rajagopal, R. Subramanian, E. Ricci, R. L. Vieri, O. Lanz, and N. Sebe. 2014. Exploring transfer learning approaches for head pose classification from multi-view surveillance images. *IJCV*, 109(1–2): 146–167. DOI: [10.1007/s11263-013-0692-2](#). [57](#)
- U. Rajashekar, I. van der Linde, A. C. Bovik, and L. K. Cormack. 2008. GAFFE: A gaze-attentive fixation finding engine. *IEEE Transactions on Image Processing*, 17(4): 564–573. DOI: [10.1109/TIP.2008.917218](#). [231](#)
- S. Rane and M. Barni. 2011. Special session on secure signal processing. *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (co-chaired)*, pp. 5848–5871. [77](#)
- J. Rasmussen, 2011. Temporal point processes: The conditional intensity function. <http://people.math.aau.dk/~jgr/teaching/punktproc11/tpp.pdf>. [206](#)
- J. G. Rasmussen. 2013. Bayesian inference for Hawkes processes. *Methodology and Computing in Applied Probability*, 15(3): 623–642. DOI: [10.1007/s11009-011-9272-5](#). [206](#), [208](#)
- A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. 2014. CNN features off-the-shelf: An astounding baseline for recognition. In *CVPR Workshop*, pp. 512–519. DOI: [10.1109/CVPRW.2014.131](#). [9](#)
- R. Reiter. 2001. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press. [165](#)
- E. Ricci, G. Zen, N. Sebe, and S. Messelodi. 2013. A prototype learning framework using EMD: Application to complex scenes analysis. *IEEE TPAMI*, 35(3): 513–526. DOI: [10.1109/TPAMI.2012.131](#). [55](#)

- E. Ricci, J. Varadarajan, R. Subramanian, S. Rota Buló, N. Ahuja, and O. Lanz. 2015. Uncovering interactions and interactors: Joint estimation of head, body orientation and f-formations from surveillance videos. In *IEEE ICCV*, pp. 4660–4668. DOI: [10.1109/ICCV.2015.529](https://doi.org/10.1109/ICCV.2015.529). 57, 70, 71, 72, 73
- L. Riungu-Kalliosaari, J. Kasurinen, and K. Smolander. 2011. Cloud services and cloud gaming in game development. *IADIS International Conference Game and Entertainment Technologies 2013 (GET 2013)*, (1). 290
- R. L. Rivest, A. Shamir, and L. Adleman. 1978. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2): 120–126. DOI: [10.1145/359340.359342](https://doi.org/10.1145/359340.359342). 79
- M.-A. Rizoiu, L. Xie, S. Sanner, M. Cebrian, H. Yu, and P. Van Hentenryck. 2017. Expecting to be HIP: Hawkes intensity processes for social media popularity. In *International Conference on World Wide Web 2017*, pp. 1–9. Perth, Australia. <http://arxiv.org/abs/1602.06033>. DOI: [10.1145/3038912.3052650](https://doi.org/10.1145/3038912.3052650). 198
- N. Robertson and I. Reid. 2006. Estimating gaze direction from low-resolution faces in video. In *ECCV*, pp. 402–415. DOI: [10.1007/11744047_31](https://doi.org/10.1007/11744047_31). 57
- A. Rohrbach, M. Rohrbach, W. Qiu, A. Friedrich, M. Pinkal, and B. Schiele. 2014. Coherent multi-sentence video description with variable level of detail. In *GCPR*, pp. 184–195. DOI: [10.1007/978-3-319-11752-2_15](https://doi.org/10.1007/978-3-319-11752-2_15). 16, 26, 27, 29
- A. Rohrbach, M. Rohrbach, and B. Schiele. 2015a. The long-short story of movie description. In *GCPR*, pp. 209–221. DOI: [10.1007/978-3-319-24947-6_17](https://doi.org/10.1007/978-3-319-24947-6_17). 28
- A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele. 2015b. A dataset for movie description. In *CVPR*, pp. 3202–3212. 26, 27, 28
- M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. 2013. Translating video content to natural language descriptions. In *ICCV*, pp. 433–440. DOI: [10.1109/ICCV.2013.61](https://doi.org/10.1109/ICCV.2013.61). 16, 29
- M. Rohrbach, A. Rohrbach, M. Regneri, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele. 2015c. Recognizing fine-grained and composite activities using hand-centric features and script data. *IJCV*, 119(3): 346–373. DOI: [10.1007/s11263-015-0851-8](https://doi.org/10.1007/s11263-015-0851-8). 26
- R. C. Rose and D. B. Paul. 1990. A hidden Markov model based keyword recognition system. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 129–132. DOI: [10.1109/ICASSP.1990.115555](https://doi.org/10.1109/ICASSP.1990.115555). 95, 98
- P. Ross. Mar. 2009. Cloud Computing’s Killer App: Gaming. *IEEE Spectrum*, 46(3): 14. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4795441>. DOI: [10.1109/MSPEC.2009.4795441](https://doi.org/10.1109/MSPEC.2009.4795441). 289
- H. A. Rowley, S. Baluja, and T. Kanade. 1998. Neural network-based face detection. *IEEE TPAMI*, 20(1): 23–38. DOI: [10.1109/34.655647](https://doi.org/10.1109/34.655647). 225
- Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. 1998. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5): 644–655. DOI: [10.1109/76.718510](https://doi.org/10.1109/76.718510). 148

- O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. 2015. Image Net large scale visual recognition challenge. *IJCV*, 115(3): 211–252. DOI: [10.1007/s11263-015-0816-y](https://doi.org/10.1007/s11263-015-0816-y). 3
- B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. 2008. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1–3): 157–173. DOI: [10.1007/s11263-007-0090-8](https://doi.org/10.1007/s11263-007-0090-8). 220
- J. Russell. 1980. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39: 1161–1178. DOI: [10.1037/h0077714](https://doi.org/10.1037/h0077714). 236
- M. Ryyanen and A. Klapuri. April 2008. Query by humming of midi and audio using locality sensitive hashing. In *ICASSP*, pp. 2249–2252. DOI: [10.1109/ICASSP.2008.4518093](https://doi.org/10.1109/ICASSP.2008.4518093). 105
- A. Sablayrolles, M. Douze, N. Usunier, and H. Jégou. 2016. How should we evaluate supervised hashing? *arXiv:1609.06753*. 121
- A. R. Sadeghi, T. Schneider, and I. Wehrenberg. 2010. Efficient privacy-preserving face recognition. In *Proceedings of the 12th International Conference on Information, Security and Cryptology*, volume 5984 of *Lecture Notes in Computer Science*, pp. 229–244. Springer. DOI: [10.1007/978-3-642-14423-3_16](https://doi.org/10.1007/978-3-642-14423-3_16). 94, 96
- S. Saeger, B. Elizalde, C. Schulze, D. Borth, B. Raj, and I. Lane. 2016, under review. Audio content descriptors. *IEEE Transactions on Audio Speech and Language Processing*. 48
- M. S. SaghaianNejadEsfahani, Y. Luo, and S.-C. Sen. 2012. Privacy protected image denoising with secret shares. In *Proceedings of the 19th IEEE International Conference on Image Processing*, pp. 253–256. DOI: [10.1109/ICIP.2012.6466843](https://doi.org/10.1109/ICIP.2012.6466843). 89, 90
- M. Saini, P. K. Atrey, S. Mehrotra, and M. S. Kankanhalli. 2012. W^3 -Privacy: Understanding what, when, and where inference channels in multi-camera surveillance video. *Springer International Journal of Multimedia Tools and Applications*, 68(17): 135–158. DOI: [10.1007/s11042-012-1207-9](https://doi.org/10.1007/s11042-012-1207-9). 75
- M. Saini, P. K. Atrey, S. Mehrotra, and M. S. Kankanhalli. 2013. Privacy aware publication of surveillance video. *Inderscience International Journal of Trust Management in Computing and Communications*, 1(1): 23–51. DOI: [10.1504/IJTMCC.2013.052523](https://doi.org/10.1504/IJTMCC.2013.052523). 92, 93
- H. Samet. 2007. *Foundations of Multidimensional and Metric Data Structures*. Elsevier. 107
- J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. 2013. Image classification with the Fisher vector: Theory and practice. *IJCV*, 105(3): 222–245. DOI: [10.1007/s11263-013-0636-x](https://doi.org/10.1007/s11263-013-0636-x). 9
- T. Sander and C. Tschudin. 1998. On software protection via function hiding. In *Information Hiding*, volume 1525 of *Lecture Notes in Computer Science*, pp. 111–123. Springer. DOI: [10.1007/3-540-49380-8_9](https://doi.org/10.1007/3-540-49380-8_9). 81
- H. Sandhawalia and H. Jégou. March 2010. Searching with expectations. In *ICASSP, Signal Processing*, pp. 1242–1245. DOI: [10.1109/ICASSP.2010.5495403](https://doi.org/10.1109/ICASSP.2010.5495403). 123, 128

- B. Sankaran. 2010. A survey of unsupervised grammar induction. Manuscript, Simon Fraser University. 47
- A. Santella and D. DeCarlo. 2004. Robust clustering of eye movement recordings for quantification of visual interest. In *Symposium on Eye Tracking Research & Applications*, pp. 27–34. DOI: [10.1145/968363.968368](https://doi.org/10.1145/968363.968368). 223
- T. S. Saponas, J. Lester, C. Hartung, S. Agarwal, and T. Kohno. 2007. Devices that tell on you: Privacy trends in consumer ubiquitous computing. In *Proceedings of the 16th Annual USENIX Security Symposium*, volume 3, pp. 55–70. 97
- F. Sarmenta. 2001. Volunteer computing. Technical report, Massachusetts Institute of Technology. 258
- N. Sarter and D. Woods. 1991. Situation awareness: A critical but ill-defined phenomenon. *The International Journal of Aviation Psychology*, 1(1): 45–57. DOI: [10.1207/s15327108ijap0101_4](https://doi.org/10.1207/s15327108ijap0101_4). 166, 167
- M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. 2009. The case for VM-based cloudlets in mobile computing. *IEEE Transactions on Pervasive Computing*, 8(4): 14–24. DOI: [10.1109/MPRV.2009.82](https://doi.org/10.1109/MPRV.2009.82). 259, 261
- M. Schneider and T. Schneider. 2014. Notes on non-interactive secure comparison in image feature extraction in the encrypted domain with privacy-preserving SIFT. In *Proceedings of the 2nd ACM Workshop on Information Hiding and Multimedia Security*, pp. 135–140. ACM. DOI: [10.1145/2600918.2600927](https://doi.org/10.1145/2600918.2600927). 87
- C. Schuldt, I. Laptev, and B. Caputo. 2004. Recognizing human actions: A local SVM approach. In *ICPR*, pp. 32–36. DOI: [10.1109/ICPR.2004.747](https://doi.org/10.1109/ICPR.2004.747). 20
- N. Sebe, I. Cohen, T. Gevers, and T. S. Huang. 2006. Emotion recognition based on joint visual and audio cues. In *International Conference on Pattern Recognition*, volume 1, pp. 1136–1139. DOI: [10.1109/ICPR.2006.489](https://doi.org/10.1109/ICPR.2006.489). 221
- M. L. Seltzer, B. Raj, and R. M. Stern. 2004. Likelihood-maximizing beamforming for robust hands-free speech recognition. *IEEE Transactions on Speech and Audio Processing*, 12(5): 489–498. DOI: [10.1109/TSA.2004.832988](https://doi.org/10.1109/TSA.2004.832988). 36
- A. Senior, S. Pankanti, A. Hampapur, L. Brown, Y.-L. Tian, and A. Ekin. 2003. Blinkering surveillance: Enabling video privacy through computer vision. Technical report, IBM. 92, 93
- F. Setti, O. Lanz, R. Ferrario, V. Murino, and M. Cristani. 2013. Multi-scale F-formation discovery for group detection. In *Proceedings of the International Conference on Image Processing (ICIP)*, pp. 3547–3551. DOI: [10.1109/ICIP.2013.6738732](https://doi.org/10.1109/ICIP.2013.6738732). 52, 69, 71
- F. Setti, C. Russell, C. Bassetti, and M. Cristani. 2015. F-formation detection: Individuating free-standing conversational groups in images. *PLoS ONE*, 10(5). DOI: [10.1371/journal.pone.0123783](https://doi.org/10.1371/journal.pone.0123783). 69, 70, 71
- H. Shacham and B. Waters. 2008. Compact proofs of retrievability. *Journal of Cryptology*, 26(3): 442–483. DOI: [10.1007/s00145-012-9129-2](https://doi.org/10.1007/s00145-012-9129-2). 260

- G. Shakhnarovich, T. Darrell, and P. Indyk. March 2006. *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*, chapter 3. MIT Press. 105
- S. Sharma, R. Kiros, and R. Salakhutdinov. 2015. Action recognition using visual attention. *arXiv:1511.04119*. 13
- J. Shashank, P. Kowshik, K. Srinathan, and C. Jawahar. 2008. Private content based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. DOI: [10.1109/CVPR.2008.4587388](https://doi.org/10.1109/CVPR.2008.4587388). 84, 86, 91, 93
- M. V. Shashanka and P. Smaragdis. 2006. Secure sound classification: Gaussian mixture models. In *2006 IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 3, pp. 1088–1091. IEEE. DOI: [10.1109/ICASSP.2006.1660847](https://doi.org/10.1109/ICASSP.2006.1660847). 99, 100
- M. Shashanka, B. Raj, and P. Smaragdis. May 2008. Probabilistic latent variable models as non-negative factorizations. *Computational Intelligence and Neuroscience*, article 947438. DOI: [10.1155/2008/947438](https://doi.org/10.1155/2008/947438). 37
- R. Shea, J. Liu, E. C. Ngai, and Y. Cui. 2013. Cloud gaming: Architecture and performance. *IEEE Network*, 27(August): 16–21. DOI: [10.1109/MNET.2013.6574660](https://doi.org/10.1109/MNET.2013.6574660). 301
- R. Shea, S. Member, D. Fu, S. Member, J. Liu, and S. Member. 2015. Cloud gaming: Understanding the support from advanced virtualization and hardware. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(12): 2026–2037. DOI: [10.1109/TCSVT.2015.2450172](https://doi.org/10.1109/TCSVT.2015.2450172). 299, 301, 302, 305
- A. Sheh and D. P. Ellis. 2003. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proceedings of the International Conference on Music Information Retrieval ISMIR-03*, <http://doi.org/10.7916/D8TB1H83>. DOI: [10.7916/D8TB1H83](https://doi.org/10.7916/D8TB1H83). 37, 38
- S. Shekhar, P. R. Schrater, R. R. Vatsavai, W. Wu, and S. Chawla. 2002. Spatial contextual classification and prediction models for mining geospatial data. *IEEE Transactions on Multimedia*, 4(2): 174–188. DOI: [10.1109/TMM.2002.1017732](https://doi.org/10.1109/TMM.2002.1017732). 186
- P. Shenoy and D. S. Tan. 2008. Human-aided computing: Utilizing implicit human processing to classify images. In *ACM Conference on Human Factors in Computing Systems*, pp. 845–854. DOI: [10.1145/1357054.1357188](https://doi.org/10.1145/1357054.1357188). 220
- B. Sheridan. March 9, 2009. A trillion points of data. *Newsweek*, 34–37. 162
- S. Shi, C.-H. Hsu, K. Nahrstedt, and R. Campbell. 2011. Using graphics rendering contexts to enhance the real-time video coding for mobile cloud gaming. *Proceedings of the 19th ACM International Conference on Multimedia*, p. 103. <http://dl.acm.org/citation.cfm?doid=2072298.2072313>. DOI: [10.1145/2072298.2072313](https://doi.org/10.1145/2072298.2072313). 307
- E. Shmueli, V. K. Singh, B. Lepri, and A. Pentland. 2014. Sensing, understanding, and shaping social behavior. *IEEE Transactions on Computational Social Systems*, 1(1): 22–34. DOI: [10.1109/TCSS.2014.2307438](https://doi.org/10.1109/TCSS.2014.2307438). 186
- T. Shortell and A. Shokoufandeh. 2015. Secure brightness/contrast filter using fully homomorphic encryption. In *Proceedings of the 14th International Conference on*

- Information Processing in Sensor Networks*, pp. 346–347. ACM. DOI: [10.1145/2737095.2742922](https://doi.org/10.1145/2737095.2742922). 89, 90
- J. Shotton, A. Fitzgibbon, A. Blake, A. Kipman, M. Finocchio, B. Moore, and T. Sharp. 2013. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1): 116–124. DOI: [10.1109/CVPR.2011.5995316](https://doi.org/10.1109/CVPR.2011.5995316). 56
- A. Shrivastava and P. Li. 2014. Asymmetric LSH for sublinear time maximum inner product search. In *NIPS*, pp. 2321–2329. 133
- H. T. Siegelmann and E. D. Sontag. 1991. Turing computability with neural nets. *Applied Mathematics Letters*, 4(6): 77–80. DOI: [10.1.1.47.8383](https://doi.org/10.1.1.47.8383). 6
- S. Siersdorfer, J. S. Pedro, and M. Sanderson. 2009. Automatic video tagging using content redundancy. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 395–402. DOI: [10.1145/1571941.1572010](https://doi.org/10.1145/1571941.1572010). 15
- K. Simonyan and A. Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 568–576. 10, 11, 12, 24
- K. Simonyan and A. Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*, pp. 1–14. 5, 9, 16
- K. Simonyan, A. Vedaldi, and A. Zisserman. 2013. Learning local feature descriptors using convex optimisation. Technical report, Department of Engineering Science, University of Oxford. 124
- R. Singh, B. Raj, and P. Smaragdis. 2010a. Latent-variable decomposition based dereverberation of monaural and multi-channel signals. In *Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 1914–1917. DOI: [10.1109/ICASSP.2010.5495326](https://doi.org/10.1109/ICASSP.2010.5495326). 37
- V. Singh and R. Agarwal. 2016. Cooperative phoneotypes: Exploring phone-based behavioral markers of cooperation. In *Proceedings of the ACM International Conference on Ubiquitous Computing*, pp. 646–657. DOI: [10.1145/2971648.2971755](https://doi.org/10.1145/2971648.2971755). 186
- V. Singh and R. Jain. 2009b. Situation based control for cyber-physical environments. In *Military Communications Conference (MILCOM) 2009. IEEE*, pp. 1–7. DOI: [10.1109/MILCOM.2009.5380000](https://doi.org/10.1109/MILCOM.2009.5380000). 166, 167, 185
- V. Singh, H. Pirsiavash, I. Rishabh, and R. Jain. 2009a. Towards environment-to-environment (E2E) multimedia communication systems. *Multimedia Tools and Applications*, 44(3): 361–388. DOI: [10.1007/s11042-009-0281-0](https://doi.org/10.1007/s11042-009-0281-0). 185
- V. Singh, M. Gao, and R. Jain. 2010b. Event analytics on microblogs. In *Proceedings of the ACM Web Science Conference*, pp. 1–4. ACM. 185
- V. Singh, M. Gao, and R. Jain. 2010c. From microblogs to social images: Event analytics for situation assessment. In *Proceedings of the International Conference on Multimedia Information Retrieval*, pp. 433–436. ACM. DOI: [10.1145/1743384.1743460](https://doi.org/10.1145/1743384.1743460). 185

- V. K. Singh and R. Jain. 2016. *Situation Recognition Using Eventshop*. Springer. 161, 162, 170, 172, 175, 177, 183, 185
- V. K. Singh, R. Jain, and M. S. Kankanhalli. 2009b. Motivating contributors in social media networks. In *Proceedings of the First SIGMM Workshop on Social Media*, pp. 11–18. ACM. DOI: [10.1145/1631144.1631149](https://doi.org/10.1145/1631144.1631149). 162
- V. K. Singh, M. Gao, and R. Jain. 2010d. Situation detection and control using spatio-temporal analysis of microblogs. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 1181–1182. ACM. DOI: [10.1145/1772690.1772864](https://doi.org/10.1145/1772690.1772864). 161, 185
- V. K. Singh, M. Gao, and R. Jain. 2010e. Social pixels: Genesis and evaluation. In *Proceedings of the 18th ACM International Conference on Multimedia*, pp. 481–490. ACM. DOI: [10.1145/1873951.1874030](https://doi.org/10.1145/1873951.1874030). 174, 183, 185
- V. K. Singh, M. Gao, and R. Jain. 2012. Situation recognition: An evolving problem for heterogeneous dynamic big multimedia data. In *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 1209–1218. ACM. DOI: [10.1145/2393347.2396421](https://doi.org/10.1145/2393347.2396421). 161, 176, 183, 185
- V. K. Singh, T.-S. Chua, R. Jain, and A. S. Pentland. 2013. Summary abstract for the 1st ACM international workshop on personal data meets distributed multimedia. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 1105–1106. ACM. DOI: [10.1145/2502081.2503836](https://doi.org/10.1145/2502081.2503836). 189
- V. K. Singh, A. Mani, and A. Pentland. 2014. Social persuasion in online and physical networks. *Proceedings of the IEEE*, 102(12): 1903–1910. DOI: [10.1109/JPROC.2014.2363986](https://doi.org/10.1109/JPROC.2014.2363986). 161, 187
- V. K. Singh, S. Pongpaichet, and R. Jain. 2016. Situation recognition from multimodal data. In *Proceedings of the 2016 ACM International Conference on Multimedia Retrieval*, pp. 1–2. ACM. DOI: [10.1145/2911996.2930061](https://doi.org/10.1145/2911996.2930061). 189
- J. Sivic and A. Zisserman. Oct. 2003. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, pp. 1470–1477. DOI: [10.1109/ICCV.2003.1238663](https://doi.org/10.1109/ICCV.2003.1238663). 114, 116
- M. Slaney, Y. Lifshits, and J. He. September 2012. Optimal parameters for locality-sensitive hashing. *Proceedings of the IEEE*, 100(9): 2604–2623. DOI: [10.1109/JPROC.2012.2193849](https://doi.org/10.1109/JPROC.2012.2193849). 115, 117
- I. Slivar, M. Suznjevic, L. Skorin-Kapov, and M. Matijasevic. Dec. 2014. Empirical QoE study of in-home streaming of online games. In *2014 13th Annual Workshop on Network and Systems Support for Games (NetGames)*, pp. 1–6. DOI: [10.1109/NetGames.2014.7010133](https://doi.org/10.1109/NetGames.2014.7010133). 296
- C. Slobogin. 2002. Public privacy: Camera surveillance of public places and the right to anonymity. *Mississippi Law Journal*, 72: 213–301. DOI: [10.2139/ssrn.364600](https://doi.org/10.2139/ssrn.364600). 91
- P. Smaragdis. 1998. Blind separation of convolved mixtures in the frequency domain. *Neurocomputing*, 22(1): 21–34. DOI: [10.1109/ASPAA.1997.625609](https://doi.org/10.1109/ASPAA.1997.625609). 36

- P. Smaragdis and B. Raj. 2010. The Markov selection model for concurrent speech recognition. In *IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 214–219. DOI: [10.1016/j.neucom.2011.09.014](https://doi.org/10.1016/j.neucom.2011.09.014). 37
- P. Smaragdis and B. Raj. 2011. Missing data imputation for time-frequency representation of audio signals. *Journal of Signal Processing Systems*, 65(3): 361–370. DOI: [10.1007/s11265-010-0512-7](https://doi.org/10.1007/s11265-010-0512-7). 37
- P. Smaragdis and B. Raj. March 2012. The Markov selection model for concurrent speech recognition. *Neurocomputing*, 80: 64–72. DOI: [10.1109/MLSP.2010.5588124](https://doi.org/10.1109/MLSP.2010.5588124). 37
- P. Smaragdis, B. Raj, and M. Shashanka. 2009a. Missing data imputation for spectral audio signals. In *IEEE International Workshop for Machine Learning in Signal Processing*, pp. 1–6. DOI: [10.1109/MLSP.2009.5306194](https://doi.org/10.1109/MLSP.2009.5306194). 37
- P. Smaragdis, M. Shashanka, and B. Raj. 2009b. A sparse non-parametric approach for single channel separation of known sounds. In *Proceedings of Neural Information Processing Systems (NIPS)*, pp. 1705–1713. 37
- A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12): 1349–1380. DOI: [10.1109/34.895972](https://doi.org/10.1109/34.895972). 85, 220
- K. Smith and P. Hancock. 1995. Situation awareness is adaptive, externally directed consciousness. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1): 137–148. DOI: [10.1518/001872095779049444](https://doi.org/10.1518/001872095779049444). 166, 167
- B. Smyth. 2007. A community-based approach to personalizing web search. *Computer*, 40(8): 42–50. DOI: [10.1109/MC.2007.259](https://doi.org/10.1109/MC.2007.259). 151
- H. Sohn, K. Plataniotis, and Y. Ro. 2010. Privacy-preserving watch list screening in video surveillance system. In *Proceedings of the 11th Pacific Rim Conference on Multimedia, Advances in Multimedia Information Processing*, volume 6297 of *Lecture Notes in Computer Science*, pp. 622–632. Springer. DOI: [10.1007/978-3-642-15702-8_57](https://doi.org/10.1007/978-3-642-15702-8_57). 94, 96
- M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic. 2012. A multimodal database for affect recognition and implicit tagging. *IEEE Trans. Affective Computing*, 3: 42–55. DOI: [10.1109/T-AFCC.2011.25](https://doi.org/10.1109/T-AFCC.2011.25). 221, 242, 243
- O. Soliman, A. Rezgui, H. Soliman, and N. Manea. 2013. Mobile cloud gaming: Issues and challenges. In F. Daniel, G. Papadopoulos, and P. Thiran, eds., *Mobile Web Information Systems SE - 10*, volume 8093 of *Lecture Notes in Computer Science*, pp. 121–128. Springer Berlin Heidelberg. ISBN 978-3-642-40275-3. DOI: [10.1007/978-3-642-40276-0_10](https://doi.org/10.1007/978-3-642-40276-0_10). 290, 291
- E. Solovey, P. Schermerhorn, M. Scheutz, A. Sassaroli, S. Fantini, and R. Jacob. 2012. Brainput: Enhancing interactive systems with streaming fNIRS brain input. In *ACM Conference on Human Factors in Computing Systems*, pp. 2193–2202. DOI: [10.1145/2207676.2208372](https://doi.org/10.1145/2207676.2208372). 251
- D. Song, D. Wagner, and A. Perig. 2000. Practical techniques for searches on encrypted data. In *Proceedings of IEEE Symposium on Security and Privacy*, pp. 44–55. DOI: [10.1109/SECPRI.2000.848445](https://doi.org/10.1109/SECPRI.2000.848445). 82

- Y. Song, L.-P. Morency, and R. Davis. 2012. Multimodal human behavior analysis: Learning correlation and interaction across modalities. In *ICMI*, pp. 27–30. DOI: [10.1145/2388676.2388684](https://doi.org/10.1145/2388676.2388684). 55
- Sony-Gaikai. July 2012. Cloud gaming adoption is accelerating . . . and fast! <http://www.nttcom.tv/2012/07/09/cloud-gaming-adoption-is-acceleratingand-fast/>. 287
- K. Soomro, A. R. Zamir, and M. Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. Computing Research Repository. <http://arxiv.org/corr/home>. 19, 21
- T. Spindler, C. Wartmann, L. Hovestadt, D. Roth, L. V. Gool, and A. Steffen. 2008. Privacy in video surveilled spaces. *Journal of Computer Security, IOS Press*, 16(2): 199–222. DOI: [10.1145/1501434.1501469](https://doi.org/10.1145/1501434.1501469). 91, 93
- N. Srivastava, E. Mansimov, and R. Salakhutdinov. 2015. Unsupervised learning of video representations using LSTMs. In *ICML*, pp. 843–852. 14, 24
- J. Staiano, M. Menéndez, A. Battocchi, A. De Angeli, and N. Sebe. 2012. UX_Mate: From facial expressions to UX evaluation. In *Designing Interactive Systems*, pp. 741–750. DOI: [10.1145/2317956.2318068](https://doi.org/10.1145/2317956.2318068). 251
- A. Steinberg, C. Bowman, and F. White. 1999. Revisions to the JDL data fusion model. Technical report, DTIC Document. DOI: [10.1117/12.341367](https://doi.org/10.1117/12.341367). 165, 166, 167
- G. Stenberg. 1992. Personality and the EEG: Arousal and emotional arousability. *Personality and Individual Differences*, 13: 1097–1113. DOI: [10.1016/0191-8869\(92\)90025-K](https://doi.org/10.1016/0191-8869(92)90025-K). 244
- Strategy Analytics. November 2014. Cloud gaming to reach inflection point in 2015. <http://tinyurl.com/p3z9hs2>.
- StreamMyGame. July 2012. StreamMyGame web page. <http://streammygame.com/>. 291
- C. Strelcha, A. M. Bronstein, M. M. Bronstein, and P. Fua. January 2012. LDAHash: Improved matching with smaller descriptors. *IEEE Trans. PAMI*, 34(1): 66–78. DOI: [10.1109/TPAMI.2011.103](https://doi.org/10.1109/TPAMI.2011.103). 120
- R. Subramanian, H. Katti, N. Sebe, M. Kankanhalli, and T.-S. Chua. 2010. An eye fixation database for saliency detection in images. In *ECCV*, pp. 30–43. DOI: [10.1007/978-3-642-15561-1_3](https://doi.org/10.1007/978-3-642-15561-1_3). 222
- R. Subramanian, V. Yanulevskaya, and N. Sebe. 2011. Can computers learn from humans to see better?: Inferring scene semantics from viewers’ eye movements. In *ACM Multimedia*, pp. 33–42. DOI: [10.1145/2072298.2072305](https://doi.org/10.1145/2072298.2072305). 221, 222, 223, 224, 225
- R. Subramanian, Y. Yan, J. Staiano, O. Lanz, and N. Sebe. 2013. On the relationship between head pose, social attention and personality prediction for unstructured and dynamic group interactions. In *Int’l Conference on Multimodal Interaction*, pp. 3–10. DOI: [10.1145/2522848.2522862](https://doi.org/10.1145/2522848.2522862). 219, 238, 240
- R. Subramanian, D. Shankar, N. Sebe, and D. Melcher. 2014. Emotion modulates eye movement patterns and subsequent memory for the gist and details of movie scenes. *Journal of Vision*, 14(3): 31. DOI: [10.1167/14.3.31](https://doi.org/10.1167/14.3.31). 237

- R. Subramanian, J. Wache, M. Abadi, R. Vieriu, S. Winkler, and N. Sebe. 2016. ASCERTAIN: Emotion and personality recognition using commercial sensors. *IEEE Transactions on Affective Computing*, issue 99. DOI: [10.1109/TAFFC.2016.2625250](https://doi.org/10.1109/TAFFC.2016.2625250). 237
- Y. Sugano, Y. Matsushita, and Y. Sato. 2013. Graph-based joint clustering of fixations and visual entities. *ACM Transactions on Applied Perception*, 10(2): 1–16. DOI: [10.1145/2465780.2465784](https://doi.org/10.1145/2465780.2465784). 224
- L. Sun, K. Jia, D.-Y. Yeung, and B. E. Shi. 2015. Human action recognition using factorized spatio-temporal convolutional networks. In *CVPR*, pp. 4597–4605. DOI: [10.1109/ICCV.2015.522](https://doi.org/10.1109/ICCV.2015.522). 10, 24
- Y. Sutcu, Q. Li, and N. Memon. 2007. Protecting biometric templates with sketch: Theory and practice. *IEEE Transactions on Information Forensics and Security*, 2(3): 503–512. 83
- I. Sutskever, O. Vinyals, and Q. V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pp. 3104–3112. 16
- A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A. L. Varna, S. He, M. Wu, and D. W. Oard. 2007. Confidentiality-preserving rank-ordered search. In *Proceedings of the ACM Workshop on Storage Security and Survivability*, pp. 7–12. DOI: [10.1145/1314313.1314316](https://doi.org/10.1145/1314313.1314316). 82, 85
- D. Szajda, M. Pohl, J. Owen, B. G. Lawson, and V. Richmond. 2006. Toward a practical data privacy scheme for a distributed implementation of the Smith-Waterman genome sequence comparison algorithm. In *Network and Distributed System Security Symposium (NDSS)*, pp. 253–265. 83
- C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. 2015a. Going deeper with convolutions. In *CVPR*, pp. 1–9. DOI: [10.1109/CVPR.2015.7298594](https://doi.org/10.1109/CVPR.2015.7298594). 5, 6, 9
- C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. 2015b. Rethinking the inception architecture for computer vision. *arXiv:1512.00567*. DOI: [10.1109/CVPR.2016.308](https://doi.org/10.1109/CVPR.2016.308). 6
- C. Szegedy, S. Ioffe, and V. Vanhoucke. 2017. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In *AAAI*, pp. 4278–4284. 6
- K. Takata, J. Ma, B. Apduhan, R. Huang, and N. Shiratori. 2008. Lifelog image analysis based on activity situation models using contexts from wearable multi sensors. In *International Conference on Multimedia and Ubiquitous Engineering*, pp. 160–163. IEEE. DOI: [10.1109/MUE.2008.69](https://doi.org/10.1109/MUE.2008.69). 165
- C.-C. Tan, Y.-G. Jiang, and C.-W. Ngo. 2011. Towards textually describing complex video contents with audio-visual concept classifiers. In *ACM Multimedia*, 655–658. DOI: [10.1145/2072298.2072411](https://doi.org/10.1145/2072298.2072411). 16
- J. Tang, S. Yan, R. Hong, G.-J. Qi, and T.-S. Chua. 2009. Inferring semantic concepts from community-contributed images and noisy tags. In *Proceedings of ACM Multimedia*, pp. 223–232. DOI: [10.1145/1631272.1631305](https://doi.org/10.1145/1631272.1631305). 47

- M. Tang, P. Agrawal, S. Pongpaichet, and R. Jain. 2015. Geospatial interpolation analytics for data streams in EventShop. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE. DOI: [10.1109/ICME.2015.7177513](https://doi.org/10.1109/ICME.2015.7177513). 186
- H. R. Tavakoli, A. Atyabi, A. Rantanen, S. J. Laukka, S. Nefti-Meziani, and J. Heikkilä. 2015. Predicting the valence of a scene from observers' eye movements. *PLoS ONE*, 10(9): 1–19. DOI: [10.1371/journal.pone.0138198](https://doi.org/10.1371/journal.pone.0138198). 237
- G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. 2010. Convolutional learning of spatio-temporal features. In *ECCV*, pp. 140–153. 13
- TeamViewer. July 2012. TeamViewer web page. <http://www.teamviewer.com>. DOI: [10.1007/978-3-642-15567-3_11](https://doi.org/10.1007/978-3-642-15567-3_11). 291
- J. Teevan, S. T. Dumais, and E. Horvitz. 2005. Personalizing search via automated analysis of interests and activities. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 449–456. ACM. DOI: [10.1145/1076034.1076111](https://doi.org/10.1145/1076034.1076111). 147
- J. Teevan, S. T. Dumais, and D. J. Liebling. 2008. To personalize or not to personalize: modeling queries with variation in user intent. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 163–170. ACM. DOI: [10.1145/1390334.1390364](https://doi.org/10.1145/1390334.1390364). 140
- A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo. 2006. Acoustic event detection and classification in smart-room environments: Evaluation of CHIL project systems. *IV Jornadas en Tecnologia del Habla*, 65(48): 5. 38
- J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. Mooney. 2014. Integrating language and vision to generate natural language descriptions of videos in the wild. In *Proceedings of the 25th International Conference on Computational Linguistics*, pp. 1218–1227. 28
- B. Thomee, D. A. Shamma, B. Elizalde, G. Friedland, K. Ni, D. Poland, D. Borth, and L. Li. 2016. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2): 64–73. 39
- Y. Tian, J. Srivastava, T. Huang, and N. Contractor. 2010. Social multimedia computing. *Computer*, 43(8): 27–36. DOI: [10.1109/MC.2010.188](https://doi.org/10.1109/MC.2010.188). 155
- S. Tok, M. Koyuncu, S. Dural, and F. Catikkas. 2010. Evaluation of International Affective Picture System (IAPS) ratings in an athlete population and its relations to personality. *Personality and Individual Differences*, 49(5): 461–466. DOI: [10.1016/j.paid.2010.04.020](https://doi.org/10.1016/j.paid.2010.04.020). 244, 246
- I. M. Toke. 2011. An introduction to Hawkes processes with applications to finance. Lectures Notes from Ecole Centrale Paris, BNP Paribas Chair of Quantitative Finance. 193
- A. Torabi, C. Pal, H. Larochelle, and A. Courville. 2015. Using descriptive video services to create a large data source for video annotation research. *arXiv:1503.01070*. 26, 27

- A. Torralba, R. Fergus, and Y. Weiss. June 2008. Small codes and large databases for recognition. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. DOI: [10.1109/CVPR.2008.4587633](https://doi.org/10.1109/CVPR.2008.4587633). 120
- W. A. A. Torres, N. Bhattacharjee, and B. Srinivasan. 2014. Effectiveness of fully homomorphic encryption to preserve the privacy of biometric data. In *Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services*, pp. 152–158. ACM. DOI: [10.1145/2684200.2684296](https://doi.org/10.1145/2684200.2684296). 82
- D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. 2015. C3D: Generic features for video analysis. In *ICCV*, 2(7): 8. [10](#), [16](#), [24](#)
- J. R. Troncoso-Pastoriza, S. Katzenbeisser, and M. Celik. 2007. Privacy preserving error resilient DNA searching through oblivious automata. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 519–528. ACM. DOI: [10.1145/1315245.1315309](https://doi.org/10.1145/1315245.1315309). 83
- B. L. Tseng, C.-Y. Lin, and J. R. Smith. 2004. Using MPEG-7 and MPEG-21 for personalizing video. *MultiMedia, IEEE*, 11(1): 42–52. DOI: [10.1109/MMUL.2004.1261105](https://doi.org/10.1109/MMUL.2004.1261105). 157
- V. Tudor, M. Almgren, and M. Papatriantafilou. 2015. Harnessing the unknown in advanced metering infrastructure traffic. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pp. 2204–2211. ACM. DOI: [10.1145/2695664.2695725](https://doi.org/10.1145/2695664.2695725). 103
- S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, and N. Sebe. 2016. Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In *CVPR*, pp. 2396–2404. DOI: [10.1109/CVPR.2016.263](https://doi.org/10.1109/CVPR.2016.263). 58
- P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. 2008. Machine recognition of human activities: A survey. *IEEE TPAMI*, 18(11): 1473–1488. DOI: [10.1109/TCSVT.2008.2005594](https://doi.org/10.1109/TCSVT.2008.2005594). 4
- T. Tuytelaars and C. Schmid. Oct. 2007. Vector quantizing feature space with a regular lattice. In *ICCV*, pp. 1–8. DOI: [10.1109/ICCV.2007.4408924](https://doi.org/10.1109/ICCV.2007.4408924). 116
- G. Tzanetakis and P. Cook. 2002. Musical genre classification of audio signals. *IEEE transactions on Speech and Audio Processing*, 10(5): 293–302. DOI: [10.1109/TSA.2002.800560](https://doi.org/10.1109/TSA.2002.800560). 37
- Ubitus. January 2015. Ubitus web page. <http://www.ubitus.net>. 287
- A. Ulges, M. Koch, and D. Borth. 2012. Linking visual concept detection with viewer demographics. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, p. 24. ACM. DOI: [10.1145/2324796.2324827](https://doi.org/10.1145/2324796.2324827). 147
- J. Ullman. 1983. *Principles of database systems*. W.H. Freeman & Co. New York. 175
- UltraVNC. July 2012. UltraVNC web page. <http://www.uvnc.com/>. 291
- M. Upmanyu. 2010. *Efficient Privacy Preserving Protocols for Visual Computation*. Master's thesis, IIIT Hyderabad, India. [82](#), [93](#), [96](#), [102](#)
- M. Upmanyu, A. M. Namboodiri, K. Srinathan, and C. V. Jawahar. 2009. Efficient privacy preserving video surveillance. In *Proceedings of IEEE 12th International Conference on Computer Vision*, pp. 1639–1646. DOI: [10.1109/ICCV.2009.5459370](https://doi.org/10.1109/ICCV.2009.5459370). [82](#), [91](#), [93](#), [96](#)

- R. Valenti, N. Sebe, and T. Gevers. 2009. Image saliency by isocentric curvedness and color. In *ICCV*, pp. 2185–2192. DOI: [10.1109/ICCV.2009.5459240](https://doi.org/10.1109/ICCV.2009.5459240). 222
- G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti. 2007. Scream and gunshot detection and localization for audio-surveillance systems. In *IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 21–26. IEEE. DOI: [10.1109/AVSS.2007.4425280](https://doi.org/10.1109/AVSS.2007.4425280). 38
- H. Van, F. Tran, and J. Menaud. 2009. SLA-aware virtual resource management for cloud infrastructures. In *Proceedings of IEEE International Conference on Computer and Information Technology (CIT)*, pp. 357–362. DOI: [10.1109/CIT.2009.109](https://doi.org/10.1109/CIT.2009.109). 260
- L. Vaquero and L. Merino. 2014. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review*, 44(5): 27–32. DOI: [10.1145/2677046.2677052](https://doi.org/10.1145/2677046.2677052). 257
- A. Vardy and Y. Be'ery. July 1993. Maximum likelihood decoding of the leech lattice. *IEEE Trans. Inform. Theory*, 39(4): 1435–1444. DOI: [10.1109/18.243466](https://doi.org/10.1109/18.243466). 115
- A. Vedaldi and A. Zisserman. March 2012. Efficient additive kernels via explicit feature maps. *IEEE Trans. PAMI*, 34: 480–492. DOI: [10.1109/CVPR.2010.5539949](https://doi.org/10.1109/CVPR.2010.5539949). 133
- R. Vedantam, C. Lawrence Zitnick, and D. Parikh. 2015. CIDeR: Consensus-based image description evaluation. In *CVPR*, pp. 4566–4575. DOI: [10.1109/CVPR.2015.7299087](https://doi.org/10.1109/CVPR.2015.7299087). 27
- A. Veen and F. P. Schoenberg. 2008. Estimation of space-time branching process models in seismology using an EM-type algorithm. *Journal of the American Statistical Association*, 103(482): 614–624. DOI: [10.1198/016214508000000148](https://doi.org/10.1198/016214508000000148). 200
- V. Veeriah, N. Zhuang, and G.-J. Qi. 2015. Differential recurrent neural networks for action recognition. In *ICCV*, pp. 4041–4049. 12
- S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. 2015a. Sequence to sequence—video to text. In *ICCV*, pp. 4534–4542. DOI: [10.1109/ICCV.2015.515](https://doi.org/10.1109/ICCV.2015.515). 16, 17, 28
- S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. 2015b. Translating videos to natural language using deep recurrent neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics—Human Language Technologies (NAACL HLT)*, pp. 1494–1504. 16, 17, 28
- S. Venugopalan, L. A. Hendricks, R. Mooney, and K. Saenko. 2016. Improving LSTM-based video description with linguistic knowledge mined from text. *arXiv:1604.01729*. DOI: [10.18653/v1/D16-1204](https://doi.org/10.18653/v1/D16-1204). 28
- T. Verbelen, S. Pieter, T. Filip, and D. Bart. 2012. Cloudlets: Bringing the cloud to the mobile user. In *Proceedings of ACM Workshop on Mobile Cloud Computing and Services (MCS)*, pp. 29–36. DOI: [10.1145/2307849.2307858](https://doi.org/10.1145/2307849.2307858). 259
- O. Verscheure, M. Vlachos, A. Anagnostopoulos, P. Frossard, E. Bouillet, and P. S. Yu. 2006. Finding “who is talking to whom” in VoIP networks via progressive stream clustering.

- In *Proceedings of IEEE Sixth International Conference on Data Mining*, pp. 667–677. DOI: [10.1109/ICDM.2006.72](https://doi.org/10.1109/ICDM.2006.72). 97, 98
- VideoLAN. VLC media player. Official page for VLC media player, the Open Source video framework. <http://www.videolan.org/vlc/>. 293
- C. Viedma. 2010. Mobile web mashups. www.mobilemashups.com 171
- A. Vinciarelli, A. Dielmann, S. Favre, and H. Salamin. 2009. Canal9: A database of political debates for analysis of social interactions. In *Proceedings of the International Conference on Affective Computing and Intelligent Interaction (ICACII)*, September 2009, pp. 1–4. DOI: [10.1109/ACII.2009.5349466](https://doi.org/10.1109/ACII.2009.5349466). 55
- O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. 2015. Show and tell: A neural image caption generator. In *CVPR*, pp. 3156–3164. DOI: [10.1109/TPAMI.2016.2587640](https://doi.org/10.1109/TPAMI.2016.2587640). 15
- P. Viola and M. J. Jones. 2004. Robust real-time face detection. *IJCV*, 57(2): 137–154. DOI: [10.1023/B:VISI.0000013087.49260.fb](https://doi.org/10.1023/B:VISI.0000013087.49260.fb). 225
- T. Virtanen, J. F. Gemmeke, B. Raj, and P. Smaragdis. 2015. Compositional models for audio processing: Uncovering the structure of sound mixtures. *Signal Processing Magazine, IEEE*, 32(2): 125–144. DOI: [10.1109/MSP.2013.2288990](https://doi.org/10.1109/MSP.2013.2288990). 42
- M. Voit and R. Stiefelhagen. 2010. 3D user-perspective, voxel-based estimation of visual focus of attention in dynamic meeting scenarios. In *ICMI*, DOI: [10.1145/1891903.1891966](https://doi.org/10.1145/1891903.1891966). 57
- L. von Ahn and L. Dabbish. 2004. Labeling images with a computer game. In *ACM Conference on Human Factors in Computing Systems*, pp. 319–326. DOI: [10.1145/985692.985733](https://doi.org/10.1145/985692.985733). 220
- J. Vuurens and A. Vries. 2012. Obtaining high-quality relevance judgments using crowdsourcing. *IEEE Transactions on Internet Computing*, 16(5): 20–27. DOI: [10.1109/MIC.2012.71](https://doi.org/10.1109/MIC.2012.71). 262
- J. Wache, R. Subramanian, M. K. Abadi, R. L. Vieriu, S. Winkler, and N. Sebe. 2015. Implicit Personality Profiling Based on Psycho-Physiological Responses to Emotional Videos. In *Int'l Conference on Multimodal Interaction*, pp. 239–246. DOI: [10.1145/2818346.2820736](https://doi.org/10.1145/2818346.2820736). 221, 237, 238, 239, 241, 245, 247, 249
- A. Wächter and L. T. Biegler. 2006. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1): 25–57. DOI: [10.1007/s10107-004-0559-y](https://doi.org/10.1007/s10107-004-0559-y). 211
- K. Walby. 2005. Open-street camera surveillance and governance in Canada. *Canadian Journal of Criminology and Criminal Justice/La Revue canadienne de criminologie et de justice pénale*, 47(4): 655–684. DOI: [10.3138/cjccj.47.4.655](https://doi.org/10.3138/cjccj.47.4.655). 91
- O. Walter, R. Haeb-Umbach, S. Chaudhuri, and B. Raj. 2013. Unsupervised word discovery from phonetic input using nested Pitman-Yor language modeling. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA) Workshop on Autonomous Learning*. 47

- L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus. 2013. Regularization of neural networks using DropConnect. In *ICML*, 28(3): 1058–1066. 6
- C. Wang, K. Ren, W. Lou, and J. Li. 2010. Toward publicly auditable secure cloud data storage services. *IEEE Network*, 24(4): 19–24. DOI: [10.1109/MNET.2010.5510914](https://doi.org/10.1109/MNET.2010.5510914). 260
- C. Wang, S. Chow, Q. Wang, K. Ren, and W. Lou. 2013. Privacy-preserving public auditing for secure cloud storage. *IEEE Transactions on Computers*, 62(2): 362–375. 260
- D. Wang and G. J. Brown (eds.). 2006. *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*, volume 147. Wiley Interscience. 36
- H. Wang and C. Schmid. 2013. Action recognition with improved trajectories. In *ICCV*, pp. 3551–3558. DOI: [10.1109/ICCV.2013.441](https://doi.org/10.1109/ICCV.2013.441). 10, 11
- J. Wang, S. Kumar, and S.-F. Chang. 2012a. Semi-supervised hashing for large-scale search. *IEEE Trans. PAMI*, 34(12): 2393–2406. DOI: [10.1109/TPAMI.2012.48](https://doi.org/10.1109/TPAMI.2012.48). 113
- J. Wang, H. T. Shen, J. Song, and J. Ji. 2014. Hashing for similarity search: A survey. *arXiv:1408.2927*. 120
- J. Wang, W. Liu, S. Kumar, and S.-F. Chang. Jan. 2016a. Learning to hash for indexing big data—A survey. *Proceedings of the IEEE*, 104(1): 34–57. DOI: [10.1109/JPROC.2015.2487976](https://doi.org/10.1109/JPROC.2015.2487976). 120
- L. Wang, Y. Qiao, and X. Tang. 2015. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pp. 4305–4314. DOI: [10.1109/CVPR.2015.7299059](https://doi.org/10.1109/CVPR.2015.7299059). 11
- L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. 2016b. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, pp. 20–36. DOI: [10.1007/978-3-319-46484-8_2](https://doi.org/10.1007/978-3-319-46484-8_2). 11, 24
- S. Wang and S. Dey. Nov. 2009. Modeling and characterizing user experience in a cloud server based mobile gaming approach. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pp. 1–7. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5425784>. DOI: [10.1109/GLOCOM.2009.5425784](https://doi.org/10.1109/GLOCOM.2009.5425784). 220, 309
- S. Wang and S. Dey. Apr. 2010a. Addressing response time and video quality in remote server based internet mobile gaming. *2010 IEEE Wireless Communication and Networking Conference*, 5: 1–6. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5506572>. DOI: [10.1109/WCNC.2010.5506572](https://doi.org/10.1109/WCNC.2010.5506572). 309
- S. Wang and S. Dey. Dec. 2010b. Rendering adaptation to address communication and computation constraints in cloud mobile gaming. In *Global Telecommunications Conference (GLOBECOM 2010), IEEE*, pp. 1–6. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5684144>. DOI: [10.1109/GLOCOM.2010.5684144](https://doi.org/10.1109/GLOCOM.2010.5684144). 309
- S. Wang and S. Dey. 2013. Adaptive mobile cloud computing to enable rich mobile multimedia applications. *IEEE Transactions on Multimedia*, 15(4): 870–883. DOI: [10.1109/TMM.2013.2240674](https://doi.org/10.1109/TMM.2013.2240674). 311

- X. Wang, S. Chen, and S. Jajodia. 2005. Tracking anonymous peer-to-peer VoIP calls on the internet. In *Proceedings of the ACM 12th Conference on Computer and Communications Security*, pp. 81–91. 97, 98
- X. Wang, A. Farhadi, and A. Gupta. 2016c. Actions \sim transformations. In *CVPR*, pp. 2658–2667. 11, 24
- Y. Wang. 2004. An FSM model for situation-aware mobile application software systems. In *ACM-SE 42: Proceedings of the 42nd Annual Southeast Regional Conference*, pp. 52–57. ISBN 1-58113-870-9. DOI: [10.1145/1102120.1102133](https://doi.org/10.1145/1102120.1102133). 165
- Y. Wang and M. S. Kankanhalli. 2015. Tweeting cameras for event detection. In *Proceedings of the 24th International Conference on World Wide Web*, pp. 1231–1241. ACM. DOI: [10.1145/2736277.2741634](https://doi.org/10.1145/2736277.2741634). 185
- Y. Wang and G. Mori. 2009. Human action recognition by semilattent topic models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10): 1762–1774. DOI: [10.1109/TPAMI.2009.43](https://doi.org/10.1109/TPAMI.2009.43). 187
- Y. Wang, C. von der Weth, Y. Zhang, K. H. Low, V. K. Singh, and M. Kankanhalli. 2016d. Concept based hybrid fusion of multimodal event signals. In *2016 IEEE International Symposium on Multimedia (ISM)*, pp. 14–19. IEEE. DOI: [10.1109/TPAMI.2009.43](https://doi.org/10.1109/TPAMI.2009.43). 185
- Z. Wang, L. Sun, X. Chen, W. Zhu, J. Liu, M. Chen, and S. Yang. 2012b. Propagation-based social-aware replication for social video contents. In *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 29–38. ACM. DOI: [10.1145/2393347.2393359](https://doi.org/10.1145/2393347.2393359). 153
- R. Weber and H. Blott. 1997. An approximation based data structure for similarity search. Technical report, ESPRIT Project HERMES. 109, 110
- R. Weber, H.-J. Schek, and S. Blott. 1998. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proceedings of the International Conference on Very Large DataBases*, pp. 194–205. 107, 109, 110
- WebM. April 2013. The WebM project web page. <http://www.webmproject.org>. 292
- K. Q. Weinberger and L. K. Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(Feb): 207–244. 149
- C. Weinhardt, A. Anandasivam, B. Blau, N. Borissov, T. Meinl, W. Michalk, and J. Stober. 2009. Cloud computing—A classification, business models, and research directions. *Business and Information Systems Engineering*, 1(5): 391–399. DOI: [10.1007/s12599-009-0071-2](https://doi.org/10.1007/s12599-009-0071-2). 259
- M. Weintraub. 1985. *A Theory and Computational Model of Auditory Monoaural Sound Separation*. PhD thesis, Department of Electrical Engineering, Stanford University. 37
- Y. Weiss, A. Torralba, and R. Fergus. Dec. 2009. Spectral hashing. In *NIPS*, pp. 1753–1760. 110, 112, 122, 124
- J. Wen, M. Severa, W. Zeng, M. Luttrell, and W. Jin. 2001. A format-compliant configurable encryption framework for access control of multimedia. In *2001 IEEE Fourth Workshop*

- on *Multimedia Signal Processing*, pp. 435–440. DOI: [10.1109/MMSP.2001.962772](https://doi.org/10.1109/MMSP.2001.962772). 92, 93
- J. Wen, M. Severa, W. Zeng, M. H. Luttrell, and W. Jin. 2002. A format-compliant configurable encryption framework for access control of video. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(6): 545–557. DOI: [10.1109/TCSVT.2002.800321](https://doi.org/10.1109/TCSVT.2002.800321). 92, 93
- P. Wieschollek, O. Wang, and A. Sorkine-Hornung. June 2016. Efficient large-scale approximate nearest neighbor search on the GPU. In *CVPR*, pp. 2027–2035. DOI: [10.1109/CVPR.2016.223](https://doi.org/10.1109/CVPR.2016.223). 111
- D. Willis, A. Dasgupta, and S. Banerjee. 2014. ParaDrop: A multi-tenant platform for dynamically installed third party services on home gateways. In *Proceedings of ACM SIGCOMM Workshop on Distributed Cloud Computing (DCC)*, pp. 43–48. DOI: [10.1145/2645892.2645901](https://doi.org/10.1145/2645892.2645901). 259
- J. G. Wilpon, L. R. Rabiner, C.-H. Lee, and E. R. Goldman. 1990. Automatic recognition of keywords in unconstrained speech using hidden Markov models. *Proceedings of IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(11): 1870–1878. DOI: [10.1109/29.103088](https://doi.org/10.1109/29.103088). 95, 98
- K. W. Wilson and B. Raj. 2010. Spectrogram dimensionality reduction with independence constraints. In *ICASSP*, pp. 1938–1941. DOI: [10.1109/ICASSP.2010.5495308](https://doi.org/10.1109/ICASSP.2010.5495308). 37
- K. W. Wilson, B. Raj, P. Smaragdis, and A. Divakaran. 2008. Speech denoising using nonnegative matrix factorization with priors. In *ICASSP*, pp. 4029–4032. DOI: [10.1109/ICASSP.2008.4518538](https://doi.org/10.1109/ICASSP.2008.4518538). 36
- C. V. Wright, L. Ballard, F. Monrose, and G. M. Masson. 2007. Language identification of encrypted VoIP traffic: Alejandra y Roberto or Alice and Bob. In *Proceedings of the 16th USENIX Security Symposium*, pp. 1–12. 97, 98
- C. V. Wright, L. Ballard, S. E. Coull, F. Monrose, and G. M. Masson. 2008. Spot me if you can: Uncovering spoken phrases in encrypted VoIP conversations. In *IEEE Symposium on Security and Privacy*, pp. 35–49. DOI: [10.1109/SP.2008.21](https://doi.org/10.1109/SP.2008.21). 97, 98
- B. Wu, S. Lyu, B.-G. Hu, and Q. Ji. 2015a. Multi-label learning with missing labels for image annotation and facial action unit recognition. *Pattern Recognition*, 48(7): 2279–2289. DOI: [10.1016/j.patcog.2015.01.022](https://doi.org/10.1016/j.patcog.2015.01.022). 58
- F. Wu and B. A. Huberman. Nov. 2007. Novelty and collective attention. *PNAS '07*, 104(45): 17599–17601. <http://www.pnas.org/content/104/45/17599.abstract>. DOI: [10.1073/pnas.0704916104](https://doi.org/10.1073/pnas.0704916104). 210
- J. Wu, C. Yuen, N. Cheung, J. Chen, and C. W. Chen. Dec. 2015b. Enabling adaptive high-frame-rate video streaming in mobile cloud gaming applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(12): 1988–2001. DOI: [10.1109/TCSVT.2015.2441412](https://doi.org/10.1109/TCSVT.2015.2441412). 309
- L. Wu, S. Garg, and R. Buyya. 2011. SLA-based resource allocation for Software as a Service Provider (SaaS) in cloud computing environments. In *Proceedings of IEEE/ACM*

- International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pp.195–204. DOI: [10.1109/CCGrid.2011.51](https://doi.org/10.1109/CCGrid.2011.51). 260
- L. Wu, R. Jin, and A. K. Jain. 2013. Tag completion for image retrieval. *IEEE TPAMI*, 35(3): 716–727. DOI: [10.1109/TPAMI.2012.124](https://doi.org/10.1109/TPAMI.2012.124). 58
- Z. Wu. 2014. Gaming in the cloud: One of the future entertainment. In *Interactive Multimedia Conference*, pp. 1–6. 290, 291
- Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. 2015c. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *ACM Multimedia*, pp. 461–470. DOI: [10.1145/2733373.2806222](https://doi.org/10.1145/2733373.2806222). 11, 12
- Z. Wu, Y.-G. Jiang, X. Wang, H. Ye, and X. Xue. 2016. Multi-stream multi-class fusion of deep networks for video classification. In *ACM Multimedia*, pp. 791–800. DOI: [10.1145/2964284.2964328](https://doi.org/10.1145/2964284.2964328). 7, 12, 24
- x264. July 2012. x264 web page. <http://www.videolan.org/developers/x264.html>. 292
- L. Xie, A. Natsev, J. R. Kender, M. Hill, and J. R. Smith. 2011. Visual memes in social media: Tracking real-world news in YouTube videos. In *Proceedings of the 19th ACM International Conference on Multimedia*, pp. 53–62. ACM. DOI: [10.1145/2072298.2072307](https://doi.org/10.1145/2072298.2072307). 156
- X. Xie, H. Liu, S. Goumaz, and W.-Y. Ma. 2005. Learning user interest for image browsing on small-form-factor devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 671–680. ACM. DOI: [10.1145/1054972.1055065](https://doi.org/10.1145/1054972.1055065). 148
- X. Xiong and F. De La Torre. 2013. Supervised descent method and its applications to face alignment. In *CVPR*, pp. 532–539. DOI: [10.1109/CVPR.2013.75](https://doi.org/10.1109/CVPR.2013.75). 225
- Z. Xiong, R. Radhakrishnan, and A. Divakaran. 2003. Generation of sports highlights using motion activity in combination with a common audio feature extraction framework. In *Proceedings of the 2003 International Conference on Image Processing*, volume 1, pp. 1–5. IEEE. DOI: [10.1109/ICIP.2003.1246884](https://doi.org/10.1109/ICIP.2003.1246884). 38
- H. Xu, J. Wang, Z. Li, G. Zeng, S. Li, and N. Yu. November 2011. Complementary hashing for approximate nearest neighbor search. In *ICCV*, pp. 1631–1638. DOI: [10.1109/ICCV.2011.6126424](https://doi.org/10.1109/ICCV.2011.6126424). 118
- H. Xu, S. Venugopalan, V. Ramanishka, M. Rohrbach, and K. Saenko. 2015a. A multi-scale multiple instance video description network. *arXiv:1505.05914*. 28
- J. Xu, E.-C. Chang, and J. Zhou. 2013. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. In *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, pp. 195–206. ACM. DOI: [10.1109/ICCV.2011.6126424](https://doi.org/10.1109/ICCV.2011.6126424). 95
- J. Xu, T. Mei, T. Yao, and Y. Rui. 2016. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, pp. 5288–5296. 15, 16, 27
- K. Xu, M. Song, X. Zhang, and J. Song. 2009. A cloud computing platform based on P2P. In *Proceedings of IEEE International Symposium on IT in Medicine and Education (ITIME)*, pp. 1–4. DOI: [10.1109/ITIME.2009.5236386](https://doi.org/10.1109/ITIME.2009.5236386). 259

- L. Xu, X. Guo, Y. Lu, S. Li, O. C. Au, and L. Fang. Jul. 2014. A low latency cloud gaming system using edge preserved image homography. In *2014 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. DOI: [10.1109/ICME.2014.6890279](https://doi.org/10.1109/ICME.2014.6890279). 307
- M. Xu, N. Maddage, C. Xu, M. Kankanhalli, and Q. Tian. 2003. Creating audio keywords for event detection in soccer video. In *Proceedings of the 2003 International Conference on Multimedia and Expo*, volume 2, pp. II–281. IEEE. DOI: [10.1109/ICME.2003.1221608](https://doi.org/10.1109/ICME.2003.1221608). 38
- R. Xu, C. Xiong, W. Chen, and J. J. Corso. 2015b. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, pp. 2346–2352. 16, 17
- Z. Xu, Y. Yang, and A. G. Hauptmann. 2015c. A discriminative CNN video representation for event detection. In *CVPR*, pp. 1798–1807. DOI: [10.1109/CVPR.2015.7298789](https://doi.org/10.1109/CVPR.2015.7298789). 9
- K. Yadati, H. Katti, and M. Kankanhalli. 2014. CAVVA: Computational affective video-in-video advertising. *IEEE Transactions on Multimedia*, 16(1): 15–23. DOI: [10.1109/TMM.2013.2282128](https://doi.org/10.1109/TMM.2013.2282128). 237
- M. A. Yakubu, P. K. Atrey, and N. C. Maddage. 2015. Secure audio reverberation over cloud. In *10th Annual Symposium on Information Assurance (ASIA '15)*, p. 39. 99, 100
- Y. Yan, E. Ricci, R. Subramanian, O. Lanz, and N. Sebe. 2013. No matter where you are: Flexible graph-guided multi-task learning for multi-view head pose classification under target motion. In *IEEE ICCV*, pp. 1177–1184. DOI: [10.1109/ICCV.2013.150](https://doi.org/10.1109/ICCV.2013.150). 57, 71
- Y. Yan, E. Ricci, R. Subramanian, G. Liu, and N. Sebe. 2014. Multitask linear discriminant analysis for view invariant action recognition. *IEEE Transactions on Image Processing (TIP)*, 23(12): 5599–5611. DOI: [10.1109/TIP.2014.2365699](https://doi.org/10.1109/TIP.2014.2365699). 56
- J. Yang, K. Yu, Y. Gong, and T. Huang. 2009. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 1794–1801. DOI: [10.1109/CVPR.2009.5206757](https://doi.org/10.1109/CVPR.2009.5206757). 233, 234, 235
- L. Yang and R. Jin. 2006. Distance metric learning: A comprehensive survey. *Michigan State University*, 2(2): 78. 149
- Q. Yang, M. J. Wooldridge, and H. Zha. 2015. Trailer generation via a point process-based visual attractiveness model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pp. 2198–2204. AAAI Press. ISBN 9781577357384. 193, 200
- Y. Yang, P. Cui, W. Zhu, and S. Yang. 2013. User interest and social influence based emotion prediction for individuals. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 785–788. ACM. DOI: [10.1145/2502081.2502204](https://doi.org/10.1145/2502081.2502204). 148, 150
- A. C.-C. Yao. July 1981. Should tables be sorted? *Journal of the ACM*, 28(3): 615–628. DOI: [10.1145/322261.322274](https://doi.org/10.1145/322261.322274). 108, 113
- L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. 2015a. Describing videos by exploiting temporal structure. In *ICCV*, pp. 4507–4515. DOI: [10.1109/ICCV.2015.512](https://doi.org/10.1109/ICCV.2015.512). 12, 16, 17, 28

- T. Yao, C.-W. Ngo, and S. Zhu. 2012. Predicting domain adaptivity: Redo or recycle? In *ACM Multimedia*, pp. 821–824. DOI: [10.1109/ICCV.2015.512](https://doi.org/10.1109/ICCV.2015.512). 17
- T. Yao, T. Mei, C.-W. Ngo, and S. Li. 2013. Annotation for free: Video tagging by mining user search behavior. In *ACM Multimedia*, pp. 977–986. DOI: [10.1145/2502081.2502085](https://doi.org/10.1145/2502081.2502085). 15
- T. Yao, T. Mei, and C.-W. Ngo. 2015b. Learning query and image similarities with ranking canonical correlation analysis. In *ICCV*, pp. 28–36. DOI: [10.1109/ICCV.2015.12](https://doi.org/10.1109/ICCV.2015.12). 17
- T. Yao, Y. Pan, C.-W. Ngo, H. Li, and T. Mei. 2015c. Semi-supervised domain adaptation with subspace learning for visual recognition. In *CVPR*, pp. 2142–2150. DOI: [10.1109/CVPR.2015.7298826](https://doi.org/10.1109/CVPR.2015.7298826). 17
- T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei. 2016. Boosting image captioning with attributes. *arXiv:1611.01646*. 17
- K. Yatani and K. N. Truong. 2012. Bodyscope: A wearable acoustic sensor for activity recognition. In *Proceedings of the ACM Conference on Ubiquitous Computing*, pp. 341–350. ACM. DOI: [10.1145/2370216.2370269](https://doi.org/10.1145/2370216.2370269). 52
- S. Yau and J. Liu. 2006. Hierarchical situation modeling and reasoning for pervasive computing. In *The Fourth IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, 2006, and the 2006 Second International Workshop on Collaborative Computing, Integration, and Assurance*, pp. 5–10. IEEE. DOI: [10.1109/SEUS-WCCIA.2006.25](https://doi.org/10.1109/SEUS-WCCIA.2006.25). 165, 167
- G. Ye, Y. Li, H. Xu, D. Liu, and S.-F. Chang. 2015a. Eventnet: A large scale structured concept library for complex event detection in video. In *ACM Multimedia*, pp. 471–480. DOI: [10.1145/2733373.2806221](https://doi.org/10.1145/2733373.2806221). 22
- H. Ye, Z. Wu, R.-W. Zhao, X. Wang, Y.-G. Jiang, and X. Xue. 2015b. Evaluating two-stream CNN for video classification. In *ACM ICMR*, pp. 435–442. DOI: [10.1145/2671188.2749406](https://doi.org/10.1145/2671188.2749406). 10
- S. Yi, Z. Hao, Z. Qin, and Q. Li. 2015. Fog computing: Platform and applications. In *Proceedings of IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, pp. 73–78. DOI: [10.1109/HotWeb.2015.22](https://doi.org/10.1109/HotWeb.2015.22). 257
- R. Yogachandran, R. Phan, J. Chambers, and D. Parish. 2012. Facial expression recognition in the encrypted domain based on local Fisher discriminant analysis. In *Proceedings of the IEEE Transactions on Affective Computing*, pp. 83–92. DOI: [10.1109/T-AFFC.2012.33](https://doi.org/10.1109/T-AFFC.2012.33). 88, 90
- H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, pp. 4584–4593. 15, 16, 28, 29
- Z. Yuan, J. Sang, Y. Liu, and C. Xu. 2013. Latent feature learning in social media network. In *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 253–262. ACM. DOI: [10.1145/2502081.2502284](https://doi.org/10.1145/2502081.2502284). 148

- M. Yuen, I. King, and K. Leung. 2011. A survey of crowdsourcing systems. In *Proceedings of IEEE International Conference on Social Computing (SocialCom)*, pp. 739–773. DOI: [10.1109/PASSAT/SocialCom.2011.203](https://doi.org/10.1109/PASSAT/SocialCom.2011.203). 257
- K. Yun, Y. Peng, D. Samaras, G. J. Zelinsky, and T. L. Berg. 2013. Studying relationships between human gaze, description, and computer vision. In *CVPR*, pp. 739–746. DOI: [10.1109/CVPR.2013.101](https://doi.org/10.1109/CVPR.2013.101). 220, 226
- G. Zen, B. Lepri, E. Ricci, and O. Lanz. 2010. Space speaks: Towards socially and personality aware visual surveillance. In *ACM International Workshop on Multimodal Pervasive Video Analysis*, pp. 37–42. DOI: [10.1145/1878039.1878048](https://doi.org/10.1145/1878039.1878048). 238, 240
- W. Zeng and S. Lei. 1999. Efficient frequency domain video scrambling for content access control. In *Proceedings of the Seventh ACM International Conference on Multimedia (Part 1)*, pp. 285–294. DOI: [10.1145/319463.319627](https://doi.org/10.1145/319463.319627). 92, 93
- W. Zeng and S. Lei. 2003. Efficient frequency domain selective scrambling of digital video. *IEEE Transactions on Multimedia*, 5(1): 118–129. DOI: [10.1109/TMM.2003.808817](https://doi.org/10.1109/TMM.2003.808817). 92, 93
- S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov. 2015. Exploiting image-trained CNN architectures for unconstrained video classification. In *BMVC*, 60: 1–13. DOI: [10.5244/C.29.60](https://doi.org/10.5244/C.29.60). 9, 24
- B. Zhang, L. Wang, Z. Wang, Y. Qiao, and H. Wang. 2016. Real-time action recognition with enhanced motion vector CNNs. In *CVPR*, pp. 2718–2726. 11
- H. Zhang, Z.-J. Zha, S. Yan, J. Bian, and T.-S. Chua. 2012. Attribute feedback. In *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 79–88. ACM. DOI: [10.1145/2393347.2393365](https://doi.org/10.1145/2393347.2393365). 148
- T. Zhang and C.-C. J. Kuo. 2001. Audio content analysis for online audiovisual data segmentation and classification. *IEEE Transactions on Speech and Audio Processing*, 9(4): 441–457. DOI: [10.1109/89.917689](https://doi.org/10.1109/89.917689). 38
- T. Zhang, C. Du, and J. Wang. June 2014. Composite quantization for approximate nearest neighbor search. In *ICML*, pp. 838–846. DOI: [10.1109/ICCV.2011.6126424](https://doi.org/10.1109/ICCV.2011.6126424). 131
- T. Zhang, G.-J. Qi, J. Tang, and J. Wang. June 2015a. Sparse composite quantization. In *CVPR*, pp. 4548–4556. DOI: [10.1109/CVPR.2015.7299085](https://doi.org/10.1109/CVPR.2015.7299085). 131
- Y.-Q. Zhang, W.-L. Zheng, and B.-L. Lu. 2015b. *Transfer Components Between Subjects for EEG-based Driving Fatigue Detection*, pp. 61–68. Springer. DOI: [10.1007/978-3-319-26561-2_8](https://doi.org/10.1007/978-3-319-26561-2_8). 250
- Z. Zhao, K. Hwang, and J. Villeta. 2012. Game cloud design with virtualized CPU/GPU servers and initial performance results. In *Proceedings of the 3rd Workshop on Scientific Cloud Computing*, ScienceCloud '12, pp. 23–30. ACM, New York. DOI: [10.1145/2287036.2287042](https://doi.org/10.1145/2287036.2287042). 301
- Y. Zheng, X. Yuan, X. Wang, J. Jiang, C. Wang, and X. Gui. 2015. Enabling encrypted cloud media center with secure deduplication. In *Proceedings of the 10th ACM Symposium on*

- Information, Computer and Communications Security*, pp. 63–72. ACM. DOI: [10.1145/2714576.2714628](https://doi.org/10.1145/2714576.2714628). 94, 96
- E. Zhong, B. Tan, K. Mo, and Q. Yang. 2013. User demographics prediction based on mobile data. *Pervasive and Mobile Computing*, pp. 823–837. DOI: [10.1016/j.pmcj.2013.07.009](https://doi.org/10.1016/j.pmcj.2013.07.009). 147
- M. Zhou, R. Zhang, W. Xie, W. Qian, and A. Zhou. 2010. Security and privacy in cloud computing: A survey. In *Proceedings of IEEE International Conference on Semantic Knowledge and Grid (SKG)*, pp. 105–112. DOI: [10.1109/SKG.2010.19](https://doi.org/10.1109/SKG.2010.19). 260
- C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal. 1997. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4): 550–560. DOI: [10.1145/279232.279236](https://doi.org/10.1145/279232.279236). 207
- W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao. 2016. A key volume mining deep framework for action recognition. In *CVPR*, pp. 1991–1999. DOI: [10.1109/CVPR.2016.219](https://doi.org/10.1109/CVPR.2016.219). 12, 24
- X. Zhu and A. B. Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*. Morgan and Claypool Publishers. 72
- X. Zhuang, S. Tsakalidis, S. Wu, P. Natarajan, R. Prasad, and P. Natarajan. 2011. Compact audio representation for event detection in consumer media. In *Proceedings of Interspeech*, pp. 2089–2092. 46
- J. R. Zipkin, F. P. Schoenberg, K. Coronges, and A. L. Bertozzi. 2016. Point-process models of social network interactions: Parameter estimation and missing data recovery. *European Journal of Applied Mathematics*, 27: 502–529. DOI: [10.1017/S0956792515000492](https://doi.org/10.1017/S0956792515000492). 209

Index

- 1-bit compressive sensing for sketch similarity, 121
- 1 Million Song Corpus for audition, 37–38
- 3D CNN model, 9–10
- 3DMark Ice Storm Benchmark, 305–306

- Abstraction in situations, 166, 168
- Accelerometers in multimodal analysis of social interactions, 55–56
- Acoustic backgrounds, 41
- Acoustic events, 41
- Acoustic intelligence, 32–33
- Action Control in EventShop platform, 179
- Actionable situations, 168
- Actions of audition objects, 47
- ActivityNet dataset for video classification, 22
- ActivityNet Large Scale Activity Recognition Challenge, 23
- Actuators in eco-systems, 162
- AdaPtive HFR vIdeo Streaming (APHIS) for cloud gaming, 309–310
- Additive and multiplicative homomorphism, 88–90
- Additive quantization in similarity searches, 131
- ADMM (alternating-direction method of multipliers), 63–66
- Advanced Micro Devices (AMD) for cloud gaming, 289
- AES (Advanced Encryption Standard), 98
- Affective ratings in emotion and personality type recognition, 240, 243–247

- Aggregation in situation recognition, 175–176
- Agreeableness personality dimension, 237
- Alerts in EventShop platform, 182
- AlexNet, 5, 9
- Algebraic homomorphic property in encrypted content, 101
- All-in-one software pack for GamingAnywhere, 294–295
- Alternating-direction method of multipliers (ADMM), 63–66
- AMD (Advanced Micro Devices) for cloud gaming, 289
- amtCNN (asymmetric multi-task CNN) model, 149
- Animation rendering
 - crowdsourced, 273–275
 - resource requirements, 256
- ANN (approximate nearest neighbors) strategies, 105, 107–111
- Annotations
 - eye fixations as, 226–236
 - user cues, 220
- Anti-sparse coding for sketch similarity, 126–127
- APHIS (AdaPtive HFR vIdeo Streaming) for cloud gaming, 309–310
- Applications for audition, 49
- Approximate nearest neighbors (ANN) strategies, 105, 107–111
- Approximate search algorithms, 107–108
- Architecture for video captioning, 17–18
- Arousal dimensions in emotion and

- personality type recognition, 236–237, 243–247
- Arrival times in Poisson processes, 195
- Asthma/allergy risk recommendations, 163, 183–184
- Asymmetric multi-task CNN (amtCNN) model, 149
- Asymmetric sketch similarity schemes, 123
- Attention mechanism for LSTM, 13
- Audio
 - emotion recognition, 221
 - multimodal analysis of social interactions, 55–56
 - multimodal pose estimation, 60
 - segment grouping, 43–44
- Audio processing in encrypted domain
 - audio editing quality enhancement, 99–100
 - speech/speaker recognition, 95–99
- Audition for multimedia computing
 - applications, 49
 - background, 35–39
 - Computer Audition field, 32–35
 - conclusion, 49–50
 - data for, 39–40
 - generative models of sound, 45
 - generative structure of audio, 44–45
 - grouping audio segments, 43–44
 - nature of audio data, 40–41
 - NELS, 47–48
 - overview, 31–32
 - peculiarities of sound, 41–49
 - representation and parsing of mixtures, 42–43
 - structure discovery in audio, 46–47
 - weak and opportunistic supervision, 48–49
- Augmented Lagrangian in ADMM, 64
- Augmented-reality cloud gaming, 313
- Average query time for similarity searches, 111
- B-trees in similarity searches, 107
- Bag-of-words representation
 - object recognition, 232–233
 - similarity searches, 116
- Basic linear algebra subprograms (BLAS) for similarity searches, 111
- Behavior analysis in user-multimedia interaction, 146
- Benchmarks in deep learning, 19–29
- Best-bin-first strategy in similarity searches, 107
- BGN (Boneh-Goh-Nissim) cryptosystem, 99
- BGV (Brakerski-Gentry-Vaikuntanathan) scheme, 101
- Big-five marker scale (BFMS) questionnaire for emotion and personality type recognition, 238
- Big-five model for emotion recognition, 237
- Binary regularization term in multimodal pose estimation, 62
- Biometric data, SPED for, 82–83
- Biometric recognition for image processing, 88
- BLAS (basic linear algebra subprograms) for similarity searches, 111
- BLEU@N metric for video classification, 27–29
- Blind source separation (BSS) in audition, 36
- BOINC platform, 258
- Boneh-Goh-Nissim (BGN) cryptosystem, 99
- Bottlenecks in Hawkes processes, 208–209
- Bounding box annotations in scene recognition, 228–229
- Brakerski-Gentry-Vaikuntanathan (BGV) scheme, 101
- Branching structure in Hawkes processes, 200–202, 213
- Broadcasts in situation-aware applications, 164
- BSS (blind source separation) in audition, 36
- Building blocks in situation recognition, 171
- Business models in fog computing, 261

- C-ADMM (coupled alternating direction method of multipliers), 64–66
- Caller/callee pairs of streams in speech/speaker recognition, 97
- Captioning video, 14–19, 23–29
- Capture servers for GamingAnywhere, 296
- CASA (computational auditory scene analysis), 36–37
- Cascade size in Hawkes processes, 213–215
- CBIR (content-based information retrieval), 85
- CCV (Columbia Consumer Videos) dataset, 21
- CDC (cloud data center) architecture, 80
- Cell-probe model and algorithms for similarity searches
 - description, 108
 - hash functions, 113–118
 - introduction, 113–114
 - query mechanisms, 118–120
- CF (collaborative filtering)
 - recommender systems, 151–152
 - SPED for, 82
- CG (computationally grounded) factor
 - definitions for situations, 167
- Characterization in situation recognition, 175–176
- Chi-Square in similarity searches, 133
- CHIL Acoustic Event Detection campaign, 38
- Children events in Hawkes processes, 212–213
- Chroma features in audition, 37–38
- CIDEr metric for video classification, 27–29
- Classification
 - audition, 38
 - situation recognition, 175–176
- Client-server privacy-preserving image retrieval framework, 86
- Cloud computing. *See* Fog computing
- Cloud data center (CDC) architecture, 80
- Cloud gaming, 256
 - adaptive transmission, 308–310
 - cloud deployment, 298–302
 - communication, 307–310
 - conclusion, 314
 - future paradigm, 310–314
 - GamingAnywhere, 291–298
 - hardware decoders, 306
 - interaction delay, 302–304
 - introduction, 287–289
 - multiplayer games, 310–311
 - research, 289–291
 - thin client design, 302–306
- Cloudlet servers, 259
- Clusters of offspring in Hawkes processes, 201–202
- CNNs (convolutional neural networks)
 - end-to-end architectures, 9–12
 - similarity searches, 107
 - video, 5–6
- Collaborative filtering (CF)
 - recommender systems, 151–152
 - SPED for, 82
- Color SIFT (CSIFT) features in object recognition, 234–236
- Columbia Consumer Videos (CCV) dataset, 21
- Column-wise regularization in C-ADMM, 64
- Common architecture for video captioning, 17–18
- Common principles of interactions between users and multimedia data, 155
- Communications
 - cloud gaming, 307–310
 - fog computing, 257
- Complementary hash functions, 117–118
- Complex mathematical operations in encrypted multimedia analysis, 102–103
- Complexity
 - encrypted multimedia content, 101
 - similarity searches, 109–111
- Compositional models of sound, 42–43
- Compressed-domain distance estimation in similarity searches, 127–128
- Compressed speaker recognition (CSR) systems, 97

- Compression in cloud gaming, 307–308
- Computational auditory scene analysis (CASA), 36–37
- Computational bottlenecks in Hawkes processes, 208–209
- Computational complexity in encrypted multimedia content, 101
- Computational/conditional security in encrypted multimedia content, 103
- Computational cost in similarity searches, 109
- Computationally grounded (CG) factor definitions for situations, 167
- Computations in fog computing, 257
- Computer Audition. *See* Audition for multimedia computing
- Conditional random field (CRF) in video captioning, 17–18
- Confidentiality of data, SPED for, 83–84
- Connectivity for cloud gaming, 311
- Conscientiousness personality dimension, 237, 246
- Containers in fog computing, 261, 282–284
- Content-based information retrieval (CBIR), 85
- Content-centric computing, 138, 143
- Content delay in CrowdMAC framework, 266–267
- “Content Is Dead; Long Live Content!” panel, 142
- Content quality in Hawkes processes, 210
- Context-aware cloud gaming, 313–314
- Controlled minions in fog computing, 257
- Conversion operator in EventShop platform, 182
- Convolutional neural networks (CNNs)
 - end-to-end architectures, 9–12
 - similarity searches, 107
 - video, 5–6
- Cooperative component sharing in cloud gaming, 312
- Correctness in fog computing, 262
- Cosine similarity and indexing similarity searches, 133
- sketch similarity, 121–124
- Cost
 - CrowdMAC framework, 266–267
 - crowdsourced animation, 273
- “Coulda, Woulda, Shoulda: 20 Years of Multimedia Opportunities” panel, 142
- Counter-Strike game, 302
- Coupled alternating direction method of multipliers (C-ADMM), 64–66
- CRF (conditional random field) in video captioning, 17–18
- Crowdedness monitors in fog computing, 284
- CrowdMAC framework, 263–267
- Crowdsensing in SAIS, 268
- Crowdsourced animation rendering services, 273–275
- Cryptology. *See* Encrypted domain multimedia content analysis
- CSIFT features in object recognition, 234–236
- CSR (compressed speaker recognition) systems, 97
- CubeLendar system, 188
- D-CASE (Detection and Classification of Acoustic Scenes and Events) challenge, 38
- DAGs (direct acyclic graphs) in fog computing, 276–278
- Data Box project, 188
- Data-centric computing, 137–138, 143
- Data collection of social media, 144–145
- Data compression in cloud gaming, 307–308
- Data ingestion
 - EventShop platform, 179, 181
 - situation recognition, 173
- Data overhead in encrypted domain multimedia content analysis, 101
- Data representation and analysis multimedia analysis, 154
- situation recognition, 174, 185–186

- Data-source Panel in EventShop platform, 178
- Data unification in situation recognition, 173
- Datasets in video classification, 19–23
- DBNs (deep belief networks) in convolutional neural networks, 5
- DCT sign correlation for images, 88
- dE-mages, 185
- Decoders in cloud gaming, 306
- Decomposition in Hawkes processes, 204–205
- Deep belief networks (DBNs) in convolutional neural networks, 5
- Deep learning
 - benchmarks and challenges, 19–29
 - conclusion, 29
 - introduction, 3–4
 - modules, 4–8
 - video captioning, 14–19
 - video classification, 8–14, 19–23
- Delay bounded admission control algorithm, 266
- Delays
 - cloud gaming, 302–304
 - CrowdMAC framework, 266–267
- Demographic information inference from user-generated content, 147
- Descriptors in situation definitions, 167
- Detection and Classification of Acoustic Scenes and Events (D-CASE) challenge, 38
- Device-aware scalable applications for cloud gaming, 291
- Difference-of-Gaussian (DoG) transforms in SIFT, 86–87
- Digital watermarking, SPED for, 81
- Direct acyclic graphs (DAGs) in fog computing, 276–278
- Direct memory access (DMA) channels for cloud gaming, 300
- Discrete Fourier transformation for probabilistic cryptosystems, 80
- Distance embeddings in similarity searches, 133
- Distance metric learning, intention-oriented, 149–150
- Distributed principal component analysis, 275–280
- DMA (direct memory access) channels for cloud gaming, 300
- Docker containers for fog computing, 283–284
- DoG (Difference-of-Gaussian) transforms in SIFT, 86–87
- Dropout in AlexNet, 5
- E-mages
 - EventShop platform, 178–181
 - situation recognition, 173
- E2LSH techniques in similarity searches, 131
- ECGs (electrocardiograms) in emotion and personality type recognition, 238, 241–242
- Eco-systems for situation recognition, 162–164
- Edge effects in Hawkes processes, 208
- EEG (electroencephalogram) devices
 - emotion and personality type recognition, 238, 241–242
 - user cues, 220–221
- Efficiency
 - encrypted domain multimedia content analysis, 103
 - situation recognition, 174
 - SPED for, 84
- Efficient Task Assignment (ETA) algorithm in SAIS, 269–272
- EigenBehaviors in situation recognition, 187
- Electrocardiograms (ECGs) in emotion and personality type recognition, 238, 241–242
- Electroencephalogram (EEG) devices
 - emotion and personality type recognition, 238, 241–242

- Electroencephalogram (EEG) devices
 - (continued)*
 - user cues, 220–221
- Electronic voting, SPED for, 81
- ElGamal cryptosystem, 79
- Emerging applications in encrypted domain
 - multimedia content analysis, 102–103
- Emotion and personality type recognition
 - introduction, 236–238
 - materials and methods, 238–240
 - personality scores vs. affective ratings, 243–247
 - physiological feature extraction, 240–243
 - physiological signals, 247–250
 - user cues, 221
- Encoder-decoder LSTM, 14
- Encoding strategies in sketch similarity, 125–126
- Encrypted domain multimedia content analysis
 - audio processing, 95–100
 - conclusion, 104
 - future research and challenges, 101–104
 - image processing, 84–90
 - introduction, 75–78
 - SPED, 78–84
 - video processing in encrypted domain, 91–96
- End-to-end CNN architectures in image-based video classification, 9–12
- Energy usage in CrowdMAC framework, 266, 272
- Enhancement layer in cloud gaming, 308
- Environmental sound classification in audition, 38
- Epidemic type aftershock-sequences (ETAS) model, 209
- Equivalent counting point processes, 194
- ESP game, 220
- ETA (Efficient Task Assignment) algorithm in SAIS, 269–272
- ETAS (epidemic type aftershock-sequences) model, 209
- Ethical issues in situation recognition, 188
- Euclidean case in similarity searches, 106–107, 114–115
- Evaluation criteria for similarity searches, 109–113
- Event-driven servers for GamingAnywhere, 296
- EventNet dataset for video classification, 22
- Events, point processes for. *See* Point processes for events
- EventShop platform
 - asthma/allergy risk recommendation, 183–184
 - heterogeneous data, 180–182
 - operators, 181–182
 - overview, 177–178
 - situation-aware applications, 182–185
 - system design, 178–180
- Exact search algorithms, 107–108
- Exhaustive search algorithms, 107–108
- Exogenous events in Hawkes processes, 198
- Expectation in similarity searches, 128
- Expected number of future events in Hawkes processes, 212–213
- Expected value in Poisson processes, 194
- Exploding gradients in recurrent neural networks, 7
- Expressive power concept in situation recognition, 170
- Extraversion personality dimension, 237, 244
- Eye fixations and movements in object recognition
 - discussion, 232
 - emotion recognition, 237
 - fixation-based annotations, 232–236
 - free-viewing and visual search, 227–232
 - introduction, 226–227
 - materials and methods, 227
 - scene semantics inferences from, 222–226
 - user cues, 220
- Eysenck's personality model, 237
- F-formation detection in head and body pose estimation, 69–73

- Face detectors in fog computing, 283–284
- Face swapping in video surveillance systems, 92
- Facial expressions, differentiating via eye movements in, 224–226
- Facial landmark trajectories in emotion and personality type recognition, 242–243
- Fairness in cloud gaming, 311
- FCGs. *See* Free-standing conversational groups (FCGs)
- FCVID (Fudan-Columbia Video Dataset) dataset for video classification, 22
- Feature extraction in image processing, 86–88
- Feed-forward neural networks (FFNNs), 6
- Filtering
 - probabilistic cryptosystems, 80
 - recommender systems, 151–152
 - situation recognition, 175
 - SPED for, 82
- Fisher Vector encoding with Variational AutoEncoder (FV-VAE), 9
- Five-factor model for emotion recognition, 237
- Fixed-base annotations for object recognition, 232–236
- Flickr videos
 - in audition, 39
 - user favorite behavior patterns, 148
- Fog computing
 - challenges, 260–262
 - conclusion, 285–286
 - CrowdMAC framework, 263–267
 - crowdsourced animation rendering service, 273–275
 - introduction, 255–258
 - open-source platforms, 280–285
 - related work, 258–260
 - scalable and distributed principal component analysis, 275–280
 - Smartphone-Augmented Infrastructure Sensing, 268–272
- Fraud in fog computing, 262
- Frechet distances in similarity searches, 133
- Free-standing conversational groups (FCGs), 51
 - conclusion, 73–74
 - F-formation detection, 69–73
 - head and body pose estimation, 56–57, 66–69
 - introduction, 52–55
 - matrix completion for multimodal pose estimation, 59–66
 - matrix completion overview, 57–58
 - multimodal analysis of social interactions, 55–56
 - SALSA dataset, 58–59
- Free-viewing (FV) tasks
 - eye movements, 227–232
 - scene recognition, 226
- Freelance minions in fog computing, 257
- Fudan-Columbia Video Dataset (FCVID) dataset for video classification, 22
- Fully homomorphic encryption (FHE) techniques, 88–90
- Function hiding, SPED for, 81
- Future actions (FA) in situation definitions, 166
- Future events in Hawkes processes, 212–213
- FV-VAE (Fisher Vector encoding with Variational AutoEncoder), 9
- G-cluster cloud gaming company, 290
- Galvanic skin response (GSR) in emotion and personality type recognition, 238, 241–242
- Games as a Service (GaaS), 291
- GamingAnywhere
 - community participation, 297–298
 - environment setup, 294–295
 - execution, 296–297
 - introduction, 291–292
 - research, 297
 - system architecture, 293–294
 - target users, 292–293
- Gaussian distribution in situation recognition, 185

- Gaussian mixture model (GMM)
 - CSR systems, 97–99
 - speech recognition, 36
- Generative models
 - Hawkes processes, 213–215
 - sound, 45
- Generative structure of audio, 44–45
- Geolocation in audition, 39
- GIST descriptors in similarity searches, 132
- GMM (Gaussian mixture model)
 - CSR systems, 97–99
 - speech recognition, 36
- Goal based (GB) factor in situation
 - definitions, 166
- Google search engine, 151
- GoogLeNet, 6, 9
- GPUs (graphical processing units) for cloud gaming, 298–302
- Gradients in recurrent neural networks, 7
- Graph-based approaches for similarity searches, 132–134
- Graph-cut approach in F-formation detection, 69–70
- Graphical processing units (GPUs) for cloud gaming, 298–302
- Graphics compression in cloud gaming, 307–308
- Graphs in social graph modeling, 145
- Grouping audio segments, 43–44
- GSR (galvanic skin response) in emotion and personality type recognition, 238, 241–242

- Hamming space and embedding
 - similarity searches, 114–115, 132
 - sketch similarity, 121–122
- Hardware decoders in cloud gaming, 306
- Hash functions
 - similarity searches, 113–118
 - sketch similarity, 123–124
- Hawkes processes, 197
 - branching structure, 200–202
 - conclusion, 217–218
 - estimating, 211–212
 - expected number of future events, 212–213
 - generative model, 213–215
 - hands-on tutorial, 215–217
 - Hawkes model for social media, 209–217
 - information diffusion, 210–211
 - intensity function, 198–200
 - introduction, 192–193
 - likelihood function, 205–206
 - maximum likelihood estimation, 207–209, 211–212
 - parameter estimates, 205–209
 - sampling by decomposition, 204–205
 - self-exciting processes, 197–198
 - simulating events, 202–205
 - thinning algorithm, 202–204
- Head and body pose estimates (HBPE)
 - experiments, 66–69
 - F-formation detection, 69–73
 - free-standing conversational groups, 53–54
 - overview, 56–57
 - SALSA dataset, 59
- Heart rate in emotion and personality type recognition, 238, 241–243
- Heterogeneous data
 - EventShop platform, 180–182
 - user-multimedia interaction, 146
- Hidden Markov models (HMM) in speech/speaker recognition, 95–98
- Hierarchical structure in audition, 46–47
- High-intensity facial expressions in eye movements, 224–226
- High-rate quantization theory in similarity searches, 117
- Histogram of oriented gradient (HOG) descriptors, 59
- HMDB51 dataset for video classification, 21, 24
- HMM (Hidden Markov models) in speech/speaker recognition, 95–98
- HOG (Histogram of oriented gradient) descriptors, 59

- Hollywood Human Action dataset for video classification, 21
- Homomorphic encryption
 - image search and retrieval, 86
 - SPED, 79
 - speech/speaker recognition, 99
 - video processing in encrypted domain, 94
- Homophily hypothesis, 154
- Honest-but-curious model, 80–81
- Hotspots in mobile Internet, 264–268
- Hough voting method (HVFF-lin), 69–70
- Hough voting method multi-scale extension (HVFF-ms), 69–70
- Householder decomposition in sketch similarity, 124
- Human actuators in eco-systems, 162
- Human behavior in situation recognition, 187
- Human-centric computing, 138
- Human pose in free-standing conversational groups, 53
- Human sensors in eco-systems, 162
- HVFF-lin (Hough voting method), 69–70
- HVFF-ms (Hough voting method multi-scale extension), 69–70
- Hybrid approaches to similarity searches, 131–132
- Hybrid compression in cloud gaming, 307–308
- Hyper-diamond E8 lattices in similarity searches, 115
- Hypervisors for cloud gaming, 299
- IARPA Aladdin Please program, 38
- Image-based video classification, 9
- Image collectors in fog computing, 283–284
- Image processing in encrypted domain
 - biometric recognition, 88
 - feature extraction, 86–88
 - image search and retrieval, 84–86
 - quality enhancement, 88–90
- Image representation learning, intention-oriented, 148–149
- ImageNet, 9
- Imbalance factor (IF) in similarity searches, 116–117
- Immigrant events in Hawkes processes, 198, 200, 204, 210
- Impact sounds, 45
- In-situ sensing in SAIS, 270–271
- Independent sample-wise processing in SPED, 79–80
- Indexing schemes
 - image search and retrieval, 85
 - similarity searches, 107–109, 112, 133
- Inferring scene semantics from eye movements, 222–226
- Influence-based recommendation, 152
- Information diffusion in Hawkes processes, 210–211
- Information theoretic/unconditional security in encrypted multimedia content, 103
- Infrared detection in multimodal pose estimation, 60
- Infrastructure sensing in fog computing, 268–272
- Input/output memory management units (IOMMUs) for cloud gaming, 300
- Instance-level inferences in audition, 33
- Integrity of data
 - encrypted domain multimedia content analysis, 103–104
 - SPED for, 83
- Intensity
 - Hawkes processes, 198–200, 210
 - Poisson processes, 195
- Intention-oriented learning
 - distance metric, 149–150
 - image representation, 148–149
- Inter-arrival times in Poisson processes, 194–196
- Interaction delay in cloud gaming, 302–304
- Interactions of objects in audition, 47
- Interactive scenes, saccades in, 223
- Intermediate Query Panel in EventShop platform, 178

- Internal Storage in EventShop platform, 179
- Interoperable common principles in social media, 155
- Interpolation operator in EventShop platform, 182
- Intuitive query and mental model in situation recognition, 174
- Inverse transform sampling technique for Hawkes processes, 202–203
- IOMMUs (input/output memory management units) for cloud gaming, 300

- Job assignment algorithms in SAIS, 269
- Johnson-Lindenstrauss Lemma, 107
- Joint estimation in multimodal pose estimation, 62
- JPEG 2000 images in encrypted domains, 88

- K-means in similarity searches, 117, 127–129
- k-NN graphs for similarity searches, 109
- KD-trees in similarity searches, 107, 116–117
- Kernel function in Hawkes processes, 198–200, 204–205
- Kernel PAC (KPCA) in similarity searches, 133
- Kernelized LSH (KLSH) in similarity searches, 133
- Keyword recognizers (KWR) in speech/speaker recognition, 95–97
- Keyword searches for images, 85
- Kgraph method for similarity searches, 133
- KLSH (kernelized LSH) in similarity searches, 133
- Kodak Consumer Videos dataset for video classification, 20–21
- KPCA (kernel PAC) in similarity searches, 133
- Kronecker product in multimodal pose estimation, 62
- KTH dataset for video classification, 20
- Kubernetes platform
 - cloud applications, 256
 - fog computing, 281–283
- Kusanagi project, 290
- KVM technology in fog computing, 282
- KWR (keyword recognizers) in speech/speaker recognition, 95–97

- LabelMe database, 220
- Lagrangian in ADMM, 64
- Laplacian matrix in multimodal pose estimation, 62
- Large margin nearest neighbor (LMNN), 149–150
- Last-mile technology, 142
- Latent variable analysis (LVA) approach in audition, 37
- Lattices in similarity searches, 115
- Learned quantizers in similarity searches, 115–116
- Leech lattices in similarity searches, 115
- Legal issues in cloud gaming, 291
- LeNet-5 framework, 5
- Likelihood function in Hawkes processes, 205–206
- Limb tracking with visual-inertial sensors, 57
- Linear filtering in probabilistic cryptosystems, 80
- Local maxima in Hawkes processes, 207–208
- Local minima in Hawkes processes, 212
- Locality-Sensitive Hashing (LSH)
 - hash functions, 114–115
 - query-adaptive, 119–120
 - similarity searches, 105–106
 - sketch similarity, 122, 126–127
- Long short-term memory (LSTM)
 - modeling, 12–13
 - recurrent neural networks, 7–8
 - video classification, 17–19
- Long-term temporal dynamics modeling, 12–13
- Look-up operations in similarity searches, 110
- Low-intensity facial expressions in eye movements, 224–226

- Lower the floor concept in situation recognition, 171
- LSH. *See* Locality-Sensitive Hashing (LSH)
- LSTM (long short-term memory)
 - modeling, 12–13
 - recurrent neural networks, 7–8
 - video classification, 17–19
- LXC containers for fog computing, 282
- M-VAD (Montreal Video Annotation Dataset)
 - for video classification, 26
- Macroscopic behavior analysis, 146
- Magnetoencephalogram (MEG) signals in emotion recognition, 221
- Magnitude of influence in Hawkes processes, 210
- Mahout-PCA library for fog computing, 279–280
- Malicious model in SPED, 81
- MapReduce platform, 279–280
- Marked Hawkes processes, 210–211
- Marked memory kernel in Hawkes processes, 211
- Markov process in Hawkes processes, 204–205
- Massively multiplayer online role-playing games (MMORPGs), 310
- Matching biometric data, SPED for, 82–83
- Matrix completion (MC)
 - free-standing conversational groups, 54
 - head and body pose estimation, 57–58
 - multimodal pose estimation, 59–66
- Matrix completion for head and body pose estimation (MC-HBPE), 60–61, 66–69
- Maximum likelihood estimation in Hawkes processes, 207–209, 211–212
- McCulloch-Pitts model for convolutional neural networks, 5
- MCG-WEBV dataset for video classification, 21
- MDA (Module Deployment Algorithm), 282, 284
- Mean average precision in similarity searches, 112
- MediaEval dataset for audition, 39
- Medical data, SPED for, 83
- MEG (magnetoencephalogram) signals in emotion recognition, 221
- Mel-Frequency Cepstral Coefficients (MFCCs) in audition, 37
- Memory over time in Hawkes processes, 210
- Memory reads in similarity searches, 110–111
- Memorylessness property in Poisson processes, 195–196
- Mesoscopic behavior analysis, 146
- Meta inferences in audition, 33
- METEOR metric for video classification, 27–29
- MFCCs (Mel-Frequency Cepstral Coefficients) in audition, 37
- Microscopic behavior analysis, 146
- Microsoft Research Video Description Corpus (MSVD) dataset for video classification, 24–28
- Minions in fog computing, 257–258
- MIREX evaluations in audition, 37
- Missing and uncertain data in situation recognition, 185–186
- Mixtures in audition, 42–43
- MMORPGs (massively multiplayer online role-playing games), 310
- Mobile Internet, CrowdMAC framework for, 263–267
- Mobile offloading in fog computing, 258–259
- Modeling approach in situation recognition, 171–172
- Module Deployment Algorithm (MDA), 282, 284
- Montreal Video Annotation Dataset (M-VAD)
 - for video classification, 26
- MPEG-X standard, 156–157
- MPII Human Pose dataset for video classification, 22
- MPII Movie Description Corpus (MPII-MD)
 - for video classification, 26
- MSR Video to Text (MSR-VTT-10K) dataset
 - for video classification, 26–27

- MSVD (Microsoft Research Video Description Corpus) dataset for video classification, 24–28
- Multi-probe LSH
 - similarity searches, 118–119
 - sketch similarity, 122
- Multi-task learning approach in head and body pose estimation, 57
- Multimedia Commons initiative for audition, 39
- Multimedia data
 - social attributes for, 154–155
 - user interest modeling from, 147–148
- Multimedia fog platforms, 257–258
- Multimedia-sensed social computing, 156
- Multimodal analysis
 - encrypted domain multimedia content analysis, 102
 - free-standing conversational groups. *See* Free-standing conversational groups (FCGs)
 - sentiment, 150
 - social interactions, 55–56
- Multimodal pose estimation
 - matrix completion, 59–66
 - model, 60–63
 - optimization method, 63–66
- Multiplayer games, 310–311
- Music signals in audition, 37–38

- Natural audio, 33–34
- Near neighbors in similarity searches, 114
- Need gap vs. semantic gap, 139–141
- Neighbors in similarity searches, 106–111, 114
- NELL (Never Ending Language Learner) system, 48
- NELS (Never-Ending Sound Learner) proposal, 47–48
- Neuroticism personality dimension, 237, 244–245
- NN-descent algorithm for similarity searches, 133
- Non-Euclidean metrics for similarity searches, 132–134

- Non-homogeneous Poisson process, 195–197
- Non-interactive scenes, saccades in, 223
- Normalized deviation in crowdsourced animation rendering service, 274–275
- NP-hard problem in multimodal pose estimation, 61

- Object interactions with saccades, 222–224
- Object recognition with eye fixations as implicit annotations, 226–236
- Observed situations, 168
- Observers in GamingAnywhere, 293
- Offspring events in Hawkes processes, 200–201
- Olympic Sports dataset for video classification, 21
- OnLive gaming, 290
- Open-source platforms
 - cloud gaming systems, 292
 - fog computing, 280–285
- OpenCV package in EventShop platform, 179
- Openness personality dimension, 237, 246
- OpenPDS project, 188
- OpenStack platform
 - cloud applications, 256
 - fog computing, 281
- Operators Panel in EventShop platform, 178
- Opportunistic supervision in audition, 48–49
- Optimization techniques
 - cloud gaming, 288
 - hash functions, 117–118
 - multimodal pose estimation, 63–66
- OTT service for cloud gaming, 310

- P2P (peer-to-peer) paradigm
 - CrowdMAC framework, 265
 - fog computing, 259
- Paillier cryptosystem
 - homomorphic encryption, 79
 - SIFT, 86–87
 - speech/speaker recognition, 99

- ParaDrop for fog computing, 259
- Parsing of mixtures in audition, 42–43
- Pascal animal classes Eye Tracking (PET)
 - database, 226–228
- PASCAL Visual Object Classes for user cues, 220
- Pass-through GPUs for cloud gaming, 300
- Pattern matching in situation recognition, 175–176
- PCA (principal component analysis), 275–280
- peer-to-peer (P2P) paradigm
 - CrowdMAC framework, 265
 - fog computing, 259
- Percepts in audition, 43
- Personality scores in emotion and personality type recognition
 - vs. affective ratings, 243–247
 - description, 240
- Personality trait recognition, 247–248
- Personalization in EventShop platform, 182
- Personalized alerts in situation-aware applications, 164
- Persuading user action, 187–188
- PET (Pascal animal classes Eye Tracking)
 - database, 226–228
- Physiological signals in emotion and personality type recognition
 - experiments, 249–250
 - feature extraction, 240–243
 - materials and methods, 238–240
 - overview, 236–238
 - personality scores vs. affective ratings, 243–247
 - responses, 238–239
 - trait recognition, 247–249
- Play-as-you-go services in cloud gaming, 298
- Point processes for events
 - conclusion, 217–218
 - defining, 193–194
 - Hawkes processes. *See* Hawkes processes
 - introduction, 191–193
 - Poisson processes, 193–197
- Poisson processes
 - definition, 194–195
 - memorylessness property, 195–196
 - non-homogeneous, 196–197
 - points, 193–194
- POSIX platforms for GamingAnywhere, 295–296
- Power consumption in cloud gaming, 305–306
- Power-law kernel function for Hawkes processes, 215–216
- Pre-binarized version vectors in sketch similarity, 126
- Principal component analysis (PCA), 275–280
- Privacy
 - encryption for. *See* Encrypted domain multimedia content analysis
 - fog computing, 262
 - situation recognition, 174, 188
- Probabilistic cryptosystems, 80
- Probability density function in Poisson processes, 195
- Product quantization in similarity searches, 128–131
- Project+take sign approach for sketch similarity, 124–127
- Protocols in emotion and personality type recognition, 240
- Proximity sensors in multimodal analysis of social interactions, 55–56
- Public video surveillance systems, 91
- Quality enhancement in image processing in encrypted domain, 88–90
- Quality of Experience (QoE) metrics in cloud gaming, 288
- Quality of Service (QoS)
 - cloud gaming, 288
 - fog computing, 262
- Quantization in similarity searches, 106, 127–131
- Quantization-optimized LSH in sketch similarity, 126–127
- Quantizers in similarity searches, 115–116, 128–129

- Queries in similarity searches
 - mechanisms, 118–120
 - preparation cost, 110
 - query-adaptive LSH, 119–120
 - times, 109, 111
- Query plan trees in EventShop platform, 180
- Query Processing Engine in EventShop platform, 179–180

- R-trees in similarity searches, 107
- Raise the ceiling concept in situation recognition, 171
- Rank-ordered image searches, 85
- Rapid prototyping toolkit in situation recognition, 172
- Raspberry Pis for fog computing, 283–284
- Raw vectors in similarity searches, 112
- Real time strategy (RTS) games, 303
- Real time streaming protocol (RTSP), 293–294
- Real time transport protocol (RTP), 293–294
- Recognition problem in situation definitions, 168
- Recommendation in social-sensed multimedia, 151–152
- Rectified Linear Units (ReLU) in AlexNet, 5
- Recurrence Quantification Analysis (RQA) in scene recognition, 229–230
- Recurrent neural networks (RNNs), 6–8
- Redundant computation in fog computing, 278
- Region-of-interest rectangles, 225
- Registered Queries in EventShop platform, 178
- Reliable data collection of social media, 144–145
- ReLU (Rectified Linear Units) in AlexNet, 5
- RenderStorm cloud rendering platform, 273
- Representation and parsing of mixtures in audition, 42–43
- Request delays in CrowdMAC framework, 266–267
- Residual vectors in similarity searches, 129
- ResNet, 6

- Resource management in fog computing, 261–262
- Results Panel in EventShop platform, 178
- Revenue in CrowdMAC framework, 266–267
- RNNs (recurrent neural networks), 6–8
- Round trip time (RTT) jitter in cloud gaming, 308
- Rowe, Larry, 142
- Rowley eye detectors, 225
- RQA (Recurrence Quantification Analysis) in scene recognition, 229–230
- RSA cryptosystem, 79
- RSA-OPRF protocol, 95
- RTP (real time transport protocol), 293–294, 303
- RTSP (real time streaming protocol), 293–294

- SaaS (Software as a Service) for cloud gaming, 290
- Saccades
 - object interactions, 222–224
 - scene recognition, 229
- SAIS (Smartphone-Augmented Infrastructure Sensing), 268–272
- SALSA (Synergistic social Scene Analysis) dataset
 - free-standing conversational groups, 58–59
 - head and body pose estimation, 66–69
- SaltStack platform
 - cloud applications, 256
 - fog computing, 281–282
- Sampling by decomposition in Hawkes processes, 204–205
- Scalable and distributed principal component analysis, 275–280
- Scalable solutions, SPED for, 84
- Scalable video coded (SVC) videos, 94–95
- Scale invariant feature transform (SIFT)
 - fog computing, 275–276
 - image processing in encrypted domain, 86–87
 - object recognition with fixation-based annotations, 234–236

- similarity searches, 107, 132
- Scene understanding
 - from eye movements, 222–226
 - user cues, 220
- Searches
 - images, 84–86
 - similarity. *See* Similarity searches
 - SPED, 82
 - speech/speaker recognition, 97
 - visual, 226–232
- Secure domains for video encoding, 92–93
- Secure processing of medical data, SPED
 - for, 83
- Secure real time transport protocol (SRTP), 98
- Security
 - vs. accuracy, 102–103
 - encryption. *See* Encrypted domain multimedia content analysis
 - fog computing, 262
- Selectivity in similarity searches, 110, 112
- Self-exciting processes
 - Hawkes processes, 197–198
 - point processes, 192
- Semantic gap
 - bridging, 145
 - vs. need gap, 139–141
- Semantic inferences in audition, 33
- Semi-controlled minions in fog computing, 257
- Semi-honest model for SPED, 80–81
- Semi-supervised hierarchical structure in audition, 46
- Sensors
 - eco-systems, 162
 - fog computing, 257
- Sequence learning for video captioning, 17
- SETI@Home application, 258
- SEVIR (Socially Embedded Visual Representation Learning) approach, 149
- Shaderlight cloud rendering platform, 273
- Shallow models and approaches
 - audition, 46
 - intention-oriented image representation learning, 148–149
- Shannon Information Theory, 138
- Shape gain in similarity searches, 115, 128
- SIDL (Social-embedding Image Distance Learning) approach, 149–150
- SIFT. *See* Scale invariant feature transform (SIFT)
- Signal processing in encrypted domain (SPED), 80
 - applications, 81–83
 - background, 78–81
 - benefits, 83–84
 - introduction, 76–78
- Similarity estimation for sketches, 121–123
- Similarity searches
 - background, 106–107
 - cell-probe algorithms, 113–120
 - conclusion, 134
 - evaluation criteria, 109–113
 - hash functions, 113–118
 - hybrid approaches, 131–132
 - introduction, 105–106
 - non-Euclidean metrics and graph-based approaches, 132–134
 - quantization, 127–131
 - query mechanisms, 118–120
 - sketches and binary embeddings, 120–127
 - types, 107–108
- Simulating events in Hawkes processes, 202–205
- Situation awareness in SAIS, 268
- Situation definitions
 - data, 168–169
 - existing, 165–167
 - features, 169
 - overview, 164
 - proposed, 167–168
 - recognition problem, 168
- Situation recognition using multimodal data
 - asthma risk-based recommendations, 163
 - challenges and opportunities, 185–188

- Situation recognition using multimodal data (*continued*)
 - conclusion, 188–189
 - eco-system, 162–164
 - EventShop platform. *See* EventShop platform
 - framework, 170–177
 - individual behavior, 186–187
 - introduction, 159–161
 - missing and uncertain data, 185–186
 - persuading user action, 187–188
 - privacy and ethical issues, 188
 - situation-aware applications, 163–164
 - situation defined, 164–170
 - situation estimation, 186
 - situation evaluation, 174–176
 - situation responses, 177
 - workflow, 172–176
- Sketches and binary embeddings
 - hash function design, 123–124
 - introduction, 120–121
 - project+take sign approach, 124–127
 - similarity estimation, 121–123
 - similarity search, 120–127
- SLA in fog computing, 262
- Smartphone-Augmented Infrastructure Sensing (SAIS), 268–272
- Social attributes for users and multimedia, 154–155
- Social-embedding Image Distance Learning (SIDL) approach, 149–150
- Social graph modeling, 145
- Social interactions in multimodal analysis, 55–56
- Social knowledge on user-multimedia interactions, 143
- Social representation of multimedia data, 145
- Social-sensed multimedia computing
 - conclusion, 157
 - demographic information inference from user-generated content, 147
 - exemplary applications, 150–153
 - future directions, 153–157
- intention-oriented distance metric learning, 149–150
- intention-oriented image representation learning, 148–149
- introduction, 137–139
- MPEG-X, 156–157
- multimedia sentiment analysis, 150
- overview, 142–144
- recent advances, 146–150
- reliable data collection, 144–145
- semantic gap vs. need gap, 139–141
- social attributes for users and multimedia, 154–155
- social representation of multimedia data, 145
- social-sensed multimedia recommendation, 151–152
- social-sensed multimedia search, 151
- social-sensed multimedia summarization, 152–153
- social-sensed video communication, 153
- user interest modeling from multimedia data, 147–148
- user-multimedia interaction behavior analysis, 146
- user profiling and social graph modeling, 145
- Socially Embedded Visual Representation Learning (SEVIR) approach, 149
- Sociometric badges in SALSA dataset, 58–59
- Software as a Service (SaaS) for cloud gaming, 290
- Sound. *See* Audition for multimedia computing
- Source separation in audition, 36
- Space and time (ST) in situation definitions, 166
- Space complexity in similarity searches, 111–112
- Space-Time ARIMA (STARIMA) models, 186
- Sparse coding in object recognition, 234
- Spatial inferences in audition, 33
- Spatial pyramid histogram representation in object recognition, 232–234

- Spatio-temporal aggregation in situation recognition, 173–174
- Spatio-temporal convolutional networks, 10
- Spatio-temporal situations, 168
- sPCA algorithm, 277–279
- SPED. *See* Signal processing in encrypted domain (SPED)
- Speech/speaker recognition, 36, 95–99
- Sports-1M dataset
 - LSTM, 12
 - video classification, 22
- ST (space and time) in situation definitions, 166
- STARIMA (Space-Time ARIMA) models, 186
- Stimuli in emotion and personality type recognition, 240
- Stop-words in similarity searches, 116
- Storage in fog computing, 257
- Stream selection in situation recognition, 172–173
- Structural questions in audition, 33
- Structure discovery in audio, 46–47
- Structured quantizers in similarity searches, 115–116
- STTPoints
 - EventShop platform, 180–181
 - situation recognition, 173
- Subcritical regime in Hawkes processes, 201
- Sum-product of two signals in probabilistic cryptosystems, 80
- Super-bits in sketch similarity, 124
- Supercritical regime in Hawkes processes, 201
- Supervised deep learning for video classification, 9–13
- Surface information in audition, 47
- Surveillance systems, 91
- SVC (scalable video coded) videos, 94–95
- Synergistic social Scene Analysis (SALSA) dataset
 - free-standing conversational groups, 58–59
 - head and body pose estimation, 66–69
- System architecture in GamingAnywhere, 293–294
- System design in EventShop platform, 178–180
- TACoS Multi-Level Corpus (TACoS-ML) dataset for video classification, 26
- Tagging video, 14–19
- TDD (trajectory-pooled deep-convolutional descriptors), 11
- Template-based language model for video captioning, 16–17
- Temporal inferences in audition, 33
- Temporal segment networks, 11
- Temporary engagement in cloud gaming, 311
- Thin client design for cloud gaming, 302–306
- Thinning algorithm for Hawkes processes, 202–204
- Third-party service providers security and privacy concerns, 75
- Throughput-driven GPUs for cloud gaming, 299
- THUMOS Challenge, 23
- Time complexity in similarity searches, 109–110
- Trajectory-pooled deep-convolutional descriptors (TDD), 11
- Transductive support vector machines (TSVMs), 68
- Transferable common principles in social media, 155
- Transmission efficiency in encrypted multimedia content analysis, 104
- TRECVID 2016 Video to Text Description (TV16-VTT) dataset for video classification, 27
- TRECVID MED dataset for video classification, 21–23
- Trust-based approaches in recommender systems, 152
- TSVM (transductive support vector machines), 68

- TV16-VTT (TRECVID 2016 Video to Text Description) dataset for video classification, 27
- Tweets dataset for fog computing, 278–280
- Two-layer LSTM networks, 12
- Two-stream approach for end-to-end CNN architectures, 10–11

- UCF-101 & THUMOS-2014 dataset for video classification, 21, 24
- Unicast alerts in situation-aware applications, 164
- Unimodal approaches for wearable devices, 52
- Unitary regularization term in multimodal pose estimation, 62
- Unsupervised hierarchical structure in audition, 46–47
- Unsupervised video feature learning, 13–14
- User-centric multimedia computing, 143–144
- User cues
 - conclusion, 250–251
 - emotion and personality type recognition, 236–250
 - eye fixations as implicit annotations for object recognition, 226–236
 - introduction, 219–222
 - scene semantics inferences from eye movements, 222–226
- User-generated content, demographic information inference from, 147
- User interest modeling from multimedia data, 147–148
- User-multimedia interaction behavior analysis, 146
- User profiling, 145
- Users, social attributes for, 154–155

- VA-files (vector approximation files) for similarity searches, 109–110
- Valence dimensions in emotion and personality type recognition, 236–237, 243–247

- Vanishing gradients in recurrent neural networks, 7
- Variable bit rate (VBR) encoding in speech/speaker recognition, 98
- Vector approximation files (VA-files) for similarity searches, 109–110
- Vector ARMA (VARMA) models, 186
- Vector identifiers in similarity searches, 112
- Vector of locally aggregated descriptors (VLAD)
 - encoding, 9
 - in similarity searches, 132
- Vegas over Access Point (VoAP) algorithm in cloud gaming, 309
- VGGNet, 5–6, 9
- Video
 - convolutional neural networks, 5–6
 - introduction, 3–4
 - recurrent neural networks, 6–8
 - social-sensed communication, 153
- Video captioning
 - approaches, 16–17
 - common architecture, 17–18
 - overview, 14–15
 - problem, 15–16
 - research, 23–29
- Video classification
 - overview, 8–9
 - research, 19–23
 - supervised deep learning, 9–13
 - unsupervised video feature learning, 13–14
- Video in cloud gaming
 - compression, 307
 - decoders, 305–306
 - sharing, 311–312
- Video processing in encrypted domain
 - introduction, 91
 - making video data unrecognizable, 91–92
 - secure domains for video encoding, 92–93
 - security implementation, 93–96
- VideoLSTM, 13
- Viola-Jones face detectors, 225

- Virality in Hawkes processes, 201, 210
- Virtualization
 - cloud gaming, 298–302, 312–313
 - fog computing, 261, 282
- Visual data and inertial sensors in head and body pose estimation, 57
- Visual Objects Classes (VOC) challenge in scene recognition, 226
- Visual search (VS) tasks
 - eye movements, 227–232
 - scene recognition, 226
- Visualization
 - EventShop platform, 182
 - situation recognition, 173
- VLAD (vector of locally aggregated descriptors)
 - encoding, 9
 - in similarity searches, 132
- VoAP (Vegas over Access Point) algorithm in cloud gaming, 309
- VOC (Visual Objects Classes) challenge in scene recognition, 226
- VOC2012 dataset, 227
- Vocal sounds, 45
- Voice over Internet protocol (VoIP) traffic, 95–99
- Volunteer computing in fog computing, 258
- Voronoi diagrams for similarity searches, 109
- Voting, SPED for, 81
- Watermarking, SPED for, 81
- Weak and opportunistic supervision in audition, 48–49
- Wearable sensing devices, 52
 - head and body pose estimation, 58
 - multimodal analysis of social interactions, 55–56
 - SALSA dataset, 58
- Weizmann dataset for video classification, 20
- Windows platforms for GamingAnywhere, 295–296
- Wisdom sources in eco-systems, 162
- Wonder Shaper for fog computing, 283
- Word-of-mouth diffusion
 - Hawkes processes, 210
 - point processes, 192
- Wrappers in EventShop platform, 181
- Xen technology in fog computing, 282
- XFINITY Games, 310
- Yahoo Flickr Creative Commons 100 Million dataset (YFCC100M) for audition, 39
- YouCook dataset for video classification, 26
- Zero-knowledge watermark detection protocol, 81
- ZYNC cloud rendering platform, 273

Editor Biography

Shih-Fu Chang



Shih-Fu Chang is the Richard Dicker Professor at Columbia University, with appointments in both Electrical Engineering Department and Computer Science Department. His research is focused on multimedia information retrieval, computer vision, machine learning, and signal processing. A primary goal of his work is to develop intelligent systems that can extract rich information from the vast amount of visual data such as those emerging on the Web, collected through pervasive sensing, or available in gigantic archives. His work on content-based visual search in the early 90s—VisualSEEk and VideoQ—set the foundation of this vibrant area. Over the years, he continued to develop innovative solutions for image/video recognition, multimodal analysis, visual content ontology, image authentication, and compact hashing for large-scale indexing. His work has had major impacts in various applications like image/video search engines, online crime prevention, mobile product search, AR/VR, and brain machine interfaces.

His scholarly work can be seen in more than 350 peer-reviewed publications, many best-paper awards, more than 30 issued patents, and technologies licensed to seven companies. He was listed as the Most Influential Scholar in the field of Multimedia by Aminer in 2016. For his long-term pioneering contributions, he has been awarded the IEEE Signal Processing Society Technical Achievement Award, ACM Multimedia Special Interest Group Technical Achievement Award, Honorary Doctorate from the University of Amsterdam, the IEEE Kiyo Tomiyasu Award, and IBM Faculty Award. For his contributions to education, he received the Great Teacher Award from the Society of Columbia Graduates. He served as Chair of ACM SIGMM

(2013–2017), Chair of Columbia Electrical Engineering Department (2007–2010), the Editor-in-Chief of the IEEE Signal Processing Magazine (2006–2008), and advisor for several international research institutions and companies. In his current capacity as Senior Executive Vice Dean at Columbia Engineering, he plays a key role in the School's strategic planning, special research initiatives, international collaboration, and faculty development. He is a Fellow of the American Association for the Advancement of Science (AAAS), a Fellow of the IEEE, and a Fellow of the ACM.