

Tuomas Virtanen · Mark D. Plumbley
Dan Ellis *Editors*

Computational Analysis of Sound Scenes and Events

 Springer

Computational Analysis of Sound Scenes and Events

Tuomas Virtanen • Mark D. Plumbley • Dan Ellis
Editors

Computational Analysis of Sound Scenes and Events

 Springer

Editors

Tuomas Virtanen
Laboratory of Signal Processing
Tampere University of Technology
Tampere, Finland

Mark D. Plumbley
Centre for Vision, Speech
and Signal Processing
University of Surrey
Surrey, UK

Dan Ellis
Google Inc.
New York, NY, USA

ISBN 978-3-319-63449-4 ISBN 978-3-319-63450-0 (eBook)
DOI 10.1007/978-3-319-63450-0

Library of Congress Control Number: 2017951198

© Springer International Publishing AG 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The recent progress on machine learning and signal processing has enabled the development of technologies for automatic analysis of sound scenes and events by computational means. This has attracted several research groups and companies to investigate this new field, which has potential in several applications and also has several research challenges. This book aims to present the state-of-the-art methodology in the field, to serve as a baseline material for people wishing to enter it or to learn more about it.

We would like to thank all the authors of the chapters of this book for their excellent contributions. We gave you hard times by making several requests, many of which were quite laborious to address. We would specifically like to thank those authors who agreed to help by cross-reviewing other chapters to make this book coherent. We would also like to thank the external reviewers, Joonas Nikunen, Guangpu Huang, Benjamin Elizalde, Mikko Parviainen, Konstantinos Drossos, Sharath Adavanne, Qiang Huang, and Yong Xu.

Tampere, Finland
Surrey, UK
New York, NY, USA
June 2017

Tuomas Virtanen
Mark D. Plumbley
Dan Ellis

Contents

Part I Foundations

- 1 Introduction to Sound Scene and Event Analysis** 3
Tuomas Virtanen, Mark D. Plumbley, and Dan Ellis
- 2 The Machine Learning Approach for Analysis of Sound Scenes and Events** 13
Toni Heittola, Emre Çakır, and Tuomas Virtanen
- 3 Acoustics and Psychoacoustics of Sound Scenes and Events** 41
Guillaume Lemaitre, Nicolas Grimault, and Clara Suied

Part II Core Methods

- 4 Acoustic Features for Environmental Sound Analysis** 71
Romain Serizel, Victor Bisot, Slim Essid, and Gaël Richard
- 5 Statistical Methods for Scene and Event Classification** 103
Brian McFee
- 6 Datasets and Evaluation** 147
Annamaria Mesaros, Toni Heittola, and Dan Ellis

Part III Advanced Methods

- 7 Everyday Sound Categorization** 183
Catherine Guastavino
- 8 Approaches to Complex Sound Scene Analysis** 215
Emmanouil Benetos, Dan Stowell, and Mark D. Plumbley
- 9 Multiview Approaches to Event Detection and Scene Analysis** 243
Slim Essid, Sanjeel Parekh, Ngoc Q.K. Duong, Romain Serizel, Alexey Ozerov, Fabio Antonacci, and Augusto Sarti

Part IV Applications

10 Sound Sharing and Retrieval 279
Frederic Font, Gerard Roma, and Xavier Serra

11 Computational Bioacoustic Scene Analysis 303
Dan Stowell

12 Audio Event Recognition in the Smart Home 335
Sacha Krstulović

13 Sound Analysis in Smart Cities 373
Juan Pablo Bello, Charlie Mydlarz, and Justin Salamon

Part V Perspectives

14 Future Perspective 401
Dan Ellis, Tuomas Virtanen, Mark D. Plumbley, and Bhiksha Raj

Index 417

Contributors

Fabio Antonacci Politecnico di Milano, Milano, Italy

Juan Pablo Bello Music and Audio Research Laboratory, New York University, New York, NY, USA

Emmanouil Benetos School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Victor Bisot LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France

Emre Çakır Tampere University of Technology, Tampere, Finland

Ngoc Q.K. Duong Technicolor, Rennes, France

Dan Ellis Google Inc, New York, NY, USA

Slim Essid LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France

Frederic Font Music Technology Group (MTG), Universitat Pompeu Fabra, Barcelona, Spain

Nicolas Grimault UMR CNRS 5292, Centre de Recherche en Neurosciences de Lyon, Université Lyon 1, Lyon Cedex, France

Catherine Guastavino School of Information Studies, McGill University, Montreal, QC, Canada

Toni Heittola Tampere University of Technology, Tampere, Finland

Sacha Krstulović Audio Analytic Ltd., Cambridge, UK

Guillaume Lemaitre STMS-Ircam-CNRS-UPMC, Paris, France

Brian McFee Center for Data Science, New York University, New York, NY, USA

Annamaria Mesaros Tampere University of Technology, Tampere, Finland

Charlie Mydlarz Center for Urban Science and Progress & Music and Audio Research Laboratory, New York University, New York, NY, USA

Alexey Ozerov Technicolor, Rennes, France

Sanjeel Parekh Technicolor, Rennes, France

Mark D. Plumbley Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, Surrey, UK

Bhiksha Raj Carnegie Mellon University, Pittsburgh, PA, USA

Gaël Richard LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France

Gerard Roma Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK

Justin Salamon Center for Urban Science and Progress & Music and Audio Research Laboratory, New York University, New York, NY, USA

Augusto Sarti Politecnico di Milano, Milano, Italy

Romain Serizel Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, France

Xavier Serra Music Technology Group (MTG), Universitat Pompeu Fabra, Barcelona, Spain

Dan Stowell School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Clara Suied Institut de Recherche Biomédicale des Armées, Brétigny-sur-Orge, France

Tuomas Virtanen Laboratory of Signal Processing, Tampere University of Technology, Tampere, Finland

Part I
Foundations

Chapter 1

Introduction to Sound Scene and Event Analysis

Tuomas Virtanen, Mark D. Plumbley, and Dan Ellis

Abstract Sounds carry a great deal of information about our environments, from individual physical events to sound scenes as a whole. In recent years several novel methods have been proposed to analyze this information automatically, and several new applications have emerged. This chapter introduces the basic concepts and research problems and engineering challenges in computational environmental sound analysis. We motivate the field by briefly describing various applications where the methods can be used. We discuss the commonalities and differences of environmental sound analysis to other major audio content analysis fields such as automatic speech recognition and music information retrieval. We discuss the main challenges in the field, and give a short historical perspective of the development of the field. We also shortly summarize the role of each chapter in the book.

Keywords Sound event detection • Sound scene classification • Sound tagging • Acoustic event detection • Acoustic scene classification • Audio content analysis

1.1 Motivation

Imagine you are standing on a street corner in a city. Close your eyes: what do you hear? Perhaps some cars and buses driving on the road, footsteps of people on the pavement, beeps from a pedestrian crossing, rustling, and clunks from shopping bags and boxes, and the hubbub of talking shoppers. Your sense of hearing tells you

T. Virtanen (✉)

Laboratory of Signal Processing, Tampere University of Technology, Tampere, Finland
e-mail: tuomas.virtanen@tut.fi

M.D. Plumbley

Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, Surrey GU2 7XH, UK
e-mail: m.plumbley@surrey.ac.uk

D. Ellis

Google Inc, 111 8th Ave, New York, NY 10027, USA
e-mail: dpwe@google.com

what is happening around you, without even needing to open your eyes, and you could do the same in a kitchen as someone is making breakfast, or listening to a tennis match on the radio.

To most people, this skill of listening to everyday events and scenes is so natural that it is taken for granted. However, this is a very challenging task for computers; the creation of “machine listening” algorithms that can automatically recognize sounds events remains an open problem.

Automatic recognition of sound events and scenes would have major impact in a wide range of applications where sound or sound sensing is—or could be—involved. For example, acoustic monitoring would allow the recognition of physical events such as glass breaking (from somebody breaking into a house), a gunshot, or a car crash. In comparison to video monitoring, acoustic monitoring can be advantageous in many scenarios, since sounds travel through obstacles, is not affected by lighting conditions, and capturing sound typically consumes less power.

There exist also large amounts of multimedia material either broadcast, uploaded via social media, or in personal collections. Current indexing methods are mostly based on textual descriptions provided by contributors or users of such media collections. Such descriptions are slow to produce manually and often quite inaccurate. Methods that automatically produce descriptions of multimedia items could lead to new, more accurate search methods that are based on the content of the materials.

Computational sound analysis can also be used to endow mobile devices with context awareness. Devices such as smartphones, tablets, robots, and cars include microphones that can be used to capture audio, as well as possessing the computational capacity to analyze the signals captured. Through audio analysis, they can recognize and react to their environment. For example, if a car “hears” children yelling from behind a corner, it can slow down to avoid a possible accident. A smartphone could automatically change its ringtone to be most appropriate for a romantic dinner, or an evening in a noisy pub.

Recent activity in the scientific community such as the DCASE challenges and related workshops—including significant commercial participation—shows a growing interest in sound scene and event analysis technologies that are discussed in this book.

1.2 What is Computational Analysis of Sound Scenes and Events?

Broadly speaking, the term *sound event* refers to a specific sound produced by a distinct physical sound source, such as a car passing by, a bird singing, or a doorbell. Sound events have a single source, although as shown by the contrast between a car and its wheels and engine, defining what counts as a single source is still subjective. Sound events typically have a well-defined, brief, duration in time. By contrast,

the term *sound scene* refers to the entirety of sound that is formed when sounds from various sources, typically from a real scenario, combine to form a mixture. For example, the sound scene of a street can contain cars passing by, footsteps, people talking, etc. The sound scene in a home might contain music from radio, a dishwasher humming, and children yelling.

The overarching goal of computational analysis of sound scenes and events is extracting information from audio by computational methods. The type of information to be extracted depends on the application. However, we can sort typical sound analysis tasks into a few high-level categories. In *classification*, the goal is to categorize an audio recording into one of a set of (predefined) categories. For example, a sound scene classification system might classify audio as one of a set of categories including home, street, and office. In (event) *detection*, the goal is to locate in time the occurrences of a specific type of sound or sounds, either by finding each instance when the sound(s) happen or by finding all the temporal positions when the sound(s) are active. There are also other more specific tasks, such as estimating whether two audio recordings are from the same sound scene.

When the classes being recognized and/or detected have associated textual descriptions, the above techniques (classification and detection) can be used to construct a verbal description of an audio signal that is understandable by humans. The number of sound events or scene classes can be arbitrarily high and in principle it is possible to train classifiers or detectors for any type of sounds that might be present in an environment. In practice the number of classes or the types of sounds that can be classified is constrained by the availability of data that is used to train classifiers, and by the accuracy of the systems. The accuracy that can be achieved is affected by many factors, such as the similarity of classes to be distinguished from each other, the diversity of each class, external factors such as interfering noises, the quality and amount of training data, and the actual computational methods used.

The above vision of automatic systems producing abstract, textual descriptions is quite different from the mainstream research on computational analysis methods of a decade ago [21], where the main focus was on lower-level processing techniques such as source separation, dereverberation, and fundamental frequency estimation. Such low-level techniques are important building blocks in classification and detection systems, but they do not yet produce information that can be naturally interpreted by humans. The number of distinct sound classes handled by current classification and detection technologies is still limited, and their analysis accuracy is to be improved, but the capability of these methods to produce human-interpretable information gives them a significantly broader potential impact than more low-level processing techniques.

The core tasks of detection and classification require using several techniques related to audio signal processing and machine learning. For example, typical computational analysis systems first extract some acoustic features from the input signal, and supervised classifiers such as neural networks can be used for classification and detection. Therefore acoustic features and classifiers, as well as more complex statistical techniques for integrating evidence, and mechanisms for representing complex world knowledge, are all core tools in the computational analysis of sound scenes and events, and hence are covered in this book.

We refer to the domain of these sound analysis techniques as “everyday sounds,” by which we mean combinations of sound sources of the number and complexity typically encountered in our daily lives. Some sound events may be quite rare (it is not every day that one encounters a snake hissing, at least for most of us), but when it does occur, it is more likely to be in the context of several other simultaneous sources than in isolation.

1.3 Related Fields

While computational analysis of non-speech, non-music sound scenes and events has only recently received widespread interest, work in analysis of speech and music signals has been around for some time. For speech signals, key tasks include recognizing the sequence of words in speech (automatic speech recognition), and recognizing the identity of the person talking (speaker recognition), or which of several people may be talking at different times (speaker diarization). For music audio, key tasks include recognizing the sequence of notes being played by one or more musical instruments (automatic music transcription), identifying the *genre* (style or category) of a musical piece (genre recognition), or identifying the instruments that are being played in a musical piece (instrument recognition): these music tasks are explored in the field of *music information retrieval* (MIR).

There are parallels between the tasks that we want to achieve for general everyday sounds, and these existing tasks. For example, the task of *sound scene classification* aims to assign a single label such as “restaurant” or “park” to an audio scene, and is related to the tasks of speaker recognition (for a speech signal with a single speaker) and musical genre recognition. Similarly, the task of *audio tagging*, which aims to assign a set of tags to a clip, perhaps naming audible objects, is related to the music task of instrument recognition in a multi-instrument musical piece. Perhaps most challenging, the task of *audio event detection*, which aims to identify the audio events—and their times—within an audio signal, is related to the speech tasks of automatic speech recognition and speaker diarization, as well as the task of automatic music transcription.

Since the analysis of everyday sounds can be related to speech and music tasks, it is not surprising to find that researchers have borrowed features and methods from speech and music, just as MIR researchers borrowed methods from the speech field. For example, features based on mel-frequency cepstral coefficients (MFCCs) [3], originally developed for speech, have also been used for MIR tasks such as genre recognition [20], and subsequently for sound scene recognition. Similarly, non-negative matrix factorization (NMF), which has been used for automatic music transcription, has also been applied to sound event recognition [4].

Nevertheless, there are differences between these domains that we should be aware of. Much of the classical work in speech recognition has focused on a single speaker, with a “source-filter” model that can separate excitation from the vocal tract: the cepstral transform at the heart of MFCCs follows directly from this

assumption, but although music and speech do not fit this model, MFCCs continue to be useful in these domains. Also, music signals often consist of sounds from instruments that have been designed to have a harmonic structure, and a particular set of “notes” (frequencies), tuned, for instance, to a western 12-semitone scale; everyday sounds will not have such carefully constructed properties. So, while existing work on speech and music can provide inspiration for everyday sound analysis, we must bear in mind that speech and music processing may not have all the answers we need.

Research on systematic classification of real-world sounds stretches back to the 1990s. One of the earliest systems was the SoundFisher of Wold et al. [22] which sought to provide similarity-based access to databases of isolated sound effects by representing each clip by a fixed-size feature vector comprising perceptual features such as loudness, pitch, and brightness. Other work grew out of the needs of the fragile speech recognizers of the time to avoid being fed non-speech signals such as music [18, 19], or to provide coarse segmentation of broadcast content [24]. The rise of cheap and ubiquitous recording devices led to interest in automatic analysis of unconstrained environmental recordings such as audio life-logs [5]. The growth of online media sharing sites such as YouTube poses enormous multimedia retrieval challenges which has fueled the current wave of interest in audio content information, including formal evaluations such as TRECVID [12, 16] which pose problems such as finding all videos relevant to “Birthday Party” or “Repairing an Appliance” among hundreds of thousands of items using both audio and visual information. While image features have proven most useful, incorporating audio features gives a consistent advantage, showing their complementary value.

Image content analysis provides an interesting comparison with the challenge of everyday sound analysis. For decades, computer vision struggled with making hard classifications of things like edges and regions even in relatively constrained images. But in the past few years, tasks such as ImageNet [17], a database of 1000 images for each of 1000 object categories, have seen dramatic jumps in performance, thanks to the development of very large “deep” neural network classifiers able to take advantage of huge training sets. We are now in an era when consumer photo services can reliably provide content-based search for a seemingly unlimited vocabulary of objects from “cake” to “sunset” within unconstrained collections of user-provided photos. This raises the question: Can we do the same thing with content-based search for specific sound events within unconstrained audio recordings?

1.4 Scientific and Technical Challenges in Computational Analysis of Sound Scenes and Events

In controlled laboratory conditions where the data used to develop computational sound scene and event analysis methods matches well with the test data, it is possible to achieve relatively high accuracies in the detection and classification of sounds

[2]. There also exist commercial products that can recognize certain specific sound categories in realistic environments [10]. However, current technologies are not able to recognize a large variety of different types of sounds in realistic environments. There are several challenges in computational sound analysis.

Many of these challenges are related to the acoustics of sound scenes and events. First, the acoustic characteristics of even a single class of sounds can be highly diverse. For example in the case of class “person yelling,” the acoustics can vary enormously depending on the person who is yelling and the way in which they yell. Second, in realistic environments there can be many different types of sounds, some of whose acoustic characteristics may be very close to the target sounds. For example, the acoustics of a person yelling can be close to vocals in some background music that is present in many environments. Thirdly, an audio signal captured by a microphone is affected by the channel coupling (impulse response) between the source and microphone, which may alter the signal sufficiently to prevent matching of models developed to recognize the sound. Finally, in realistic environments there are almost always multiple sources producing sound simultaneously. The captured audio is a superposition of all the sources present, which again distorts the signal captured. In several applications of sound scene and event analysis, microphones that are used to capture audio are often significantly further away from target sources, which increases the effect of impulse responses from source to microphone as well as other sources in the environment. This situation is quite different from speech applications, where close-talk microphones are still predominantly used.

In addition to these complications related to the acoustics of sound scenes and events, there are also several fundamental challenges related to the development of computational methods. For example, if we are aiming at the development of methods able to classify and detect a large number of sounds, there is need for a taxonomy that defines the classes to be used. However, to date there is no established taxonomy for environmental sound events or scenes.

The computational methods used are heavily based on machine learning, where the parameters of a system are automatically obtained by using examples of the target (and non-target) sounds. In contrast to the situation in image classification, currently available datasets that can be used to develop computational scene and event scene analysis systems are more limited in size, diversity, and number of event instances, even though recent contributions such as AudioSet [6] have significantly reduced this gap.

1.5 About This Book

This book will provide a comprehensive description of the whole procedure for developing computational methods for sound scene and event analysis, ranging from data acquisition and labeling, designing the taxonomy used in the system, to signal processing methods for feature extraction and machine learning methods for sound recognition. The book will discuss commonalities as well as differences between

various analysis tasks, such as scene or event classification, detection, and tagging. It will also discuss advanced techniques that can take advantage of multiple microphones or other modalities. In addition to covering this kind of general methodology, the most important application domains, including multimedia information retrieval, bioacoustic scene analysis, smart homes, and smart cities, will also be covered. The book mainly focuses on presenting the computational algorithms and mathematical models behind the methods, and does not discuss specific software or hardware implementations (even though Chap. 13 discusses some possible hardware options). The methods present in the book are meant for the analysis of any everyday sounds in general. We will not discuss highly specific types of sounds such as speech or music, since analysis problems in their case are also more specific, and there already exist literature to address them [7, 13, 23].

The book is targeted for researchers, engineers, or graduate students in computer science and electrical engineering. We assume that readers will have basic knowledge of acoustics, signal processing, machine learning, linear algebra, and probability theory—although Chaps. 2 to 5 will give some background about some of the most important concepts. For those that are not yet familiar with the above topics, we recommend the following textbooks as sources of information: [9, 15], and [11] on signal processing, [14] on psychoacoustics, [1] on machine learning, and [8] on deep neural networks.

The book is divided into five parts. Part I presents the foundations of computational sound analysis systems. Chapter 2 introduces the supervised machine learning approach to sound scene and event analysis, which is the mainstream and typically the most efficient and generic approach in developing such systems. It will discuss the commonalities and differences between sound classification, detection, and tagging, and presents an example approach based on deep neural networks that can be used in all the above tasks.

Chapter 3 gives an overview of acoustics and human perception of sound events and scenes. When designing sound analysis systems it is important to have an understanding of the acoustic properties of target sounds, to support the development of the analysis methods. Knowledge about how the human auditory system processes everyday sounds is useful, and can be used to get ideas for the development of computational methods.

Part II of the book presents in detail the signal processing and machine learning methods as well as the data required for the development of computational sound analysis systems. Chapter 4 gives an overview of acoustic features that are used to represent audio signals analysis systems. Starting from representations of sound in general, it then moves from features based on signal processing towards learning features automatically from the data. The chapter also describes how to select relevant features for an analysis task, and how to temporally integrate and pool typical features extracted from short time frames.

Chapter 5 presents various pattern classification techniques that are used to map acoustic features to information about presence of each sound event or scene class. It first discusses basic concepts of supervised learning that are used in the development of such methods, and then discusses the most common discriminative and generative

classification models, including temporal modeling with hidden Markov models. The chapter also covers various models based on deep neural networks, which are currently popular in many analysis tasks. The chapter also discusses how the robustness of classifiers can be improved by various augmentation, domain adaptation, and ensemble methods.

Chapter 6 describes what kind of data—audio recordings and their annotations—are required in the development of sound analysis systems. It discusses possible ways of obtaining such material either from existing sources or by doing new recordings and annotations. It also discusses the procedures used to evaluate analysis systems as well as objective metrics used in such evaluations.

Part III of the book presents advanced topics related to categorization of sounds, analysis of complex scenes, and use of information from multiple sources. In the supervised learning approach for sound analysis which is the most typical and most powerful approach, some categorization of sounds is needed that will be used as the basis of the analysis. Chapter 7 presents various ways to categorize everyday sounds. It first discusses various theories of classification, and how new categorizations can be obtained. Then it discusses in more detail the categorization of everyday sounds, and their taxonomies and ontologies.

Chapter 8 presents approaches for the analysis of complex sound scenes consisting of multiple sound sources. It first presents a categorization of various sound analysis tasks, from scene classification to event detection, classification, and tagging. It discusses monophonic approaches that are able to estimate only one sound class at a time, as well as polyphonic approaches that enable analysis of multiple co-occurring sounds. It also discusses how contextual information can be used in sound scene and event analysis.

Chapter 9 presents multiview approaches, where data from multiple sensors are used in the analysis. These can include, for example, visual information or multiple microphones. The chapter first discusses general system architectures used in multiview analysis, and then presents how information can be fused at various system levels (features vs. classifier level). Then it discusses in detail two particularly interesting multiview cases for sound analysis: use of visual information in addition to audio and use of multiple microphones.

Part IV of the book covers selected computational sound scene and event analysis applications. Chapter 10 focuses on sound sharing and retrieval. It describes what kind of information (e.g., audio formats, licenses, metadata, features) should be taken into account when creating an audio database for this purpose. It then presents how sound retrieval can be done based on metadata, using freesound.org as an example. Finally, it presents how retrieval can be done using audio itself.

Chapter 11 presents the computational sound analysis approach to bioacoustic scene analysis. It first introduces the possible analysis tasks addressed in bioacoustics. Then it presents computational methods used in the field, including core methods such as segmentation, detection, and classification that share similarities to other fields, advanced methods such as source separation, measuring the similarity of sounds, analysis of sounds sequences, and methods for visualization and holistic soundscape analysis. The chapter also discusses how the methods can be employed at large scale, taking into account the computational complexity of the methods.

Chapter 12 focuses on sound event detection for smart home applications. It first discusses what kind of information sound can provide for these applications, and challenges such as the diversity of non-target sounds encountered and effect of audio channel. Then it discusses the user expectations of such systems, and how it affects the metrics that should be used in the development. Finally, it discusses the privacy and data protection issues of sound analysis systems.

Chapter 13 discusses the use of sound analysis in smart city applications. It first presents what kind of possibilities there are for computational sound analysis in applications such as surveillance and noise monitoring. It then discusses sound capture options based on mobile or static sensors, and the infrastructure of sound sensing networks. Then it presents various computational sound analysis results from studies focusing on urban sound environments.

Chapter 14 presents some future perspectives related to the research topic, for example, how to automatically obtain training data (both audio and labels) for the development of automatic systems. We also discuss how unlabeled data can be used in combination with active learning to improve classifiers and label data by querying users for labels. We discuss how weakly labeled data without temporal annotations can be used for developing sound event detection systems. The book concludes with a discussion of some potential future applications of the technologies.

Accompanying website of the book <http://cassebook.github.io> includes supplementary material and software implementations which facilitates practical interaction with the methods presented.

References

1. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2007)
2. Çakır, E., Parascandolo, G., Heittola, T., Huttunen, H., Virtanen, T.: Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, **25**(6), (2017)
3. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech Signal Process.* **28**, 357–366 (1980)
4. Dikmen, O., Mesaros, A.: Sound event detection using non-negative dictionaries learned from annotated overlapping events. In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (2013)
5. Ellis, D.P., Lee, K.: Accessing minimal-impact personal audio archives. *IEEE MultiMedia* **13**(4), 30–38 (2006)
6. Gemmeke, J.F., Ellis, D.P.W., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: an ontology and human-labeled dataset for audio events. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (2017)
7. Gold, B., Morgan, N., Ellis, D.: *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. Wiley, New York (2011)
8. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press, Cambridge (2016)
9. Iffachor, E., Jervis, B.: *Digital Signal Processing: A Practical Approach*, 2nd edn. Prentice Hall, Upper Saddle River (2011)

10. Krstulović, S., et al.: AudioAnalytic – Intelligent sound detection (2016). <http://www.audioanalytic.com>
11. Lyons, R.G.: Understanding Digital Signal Processing, 3rd edn. Pearson India, Harlow (2011)
12. Metzke, F., Rawat, S., Wang, Y.: Improved audio features for large-scale multimedia event detection. In: Proceedings of IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE, New York (2014)
13. Müller, M.: Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications. Springer, Cham (2015)
14. Moore, B.: An Introduction to the Psychology of Hearing, 6th edn. BRILL, Leiden (2013)
15. Oppenheim, A.V., Schaffer, R.W.: Discrete-Time Signal Processing, 3rd edn. Pearson Education Limited, Harlow (2013)
16. Pancoast, S., Akbacak, M.: Bag-of-audio-words approach for multimedia event classification. In: Proceedings of Interspeech, pp. 2105–2108 (2012)
17. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
18. Saunders, J.: Real-time discrimination of broadcast speech/music. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 2, pp. 993–996. IEEE, New York (1996)
19. Scheirer, E., Slaney, M.: Construction and evaluation of a robust multifeature speech/music discriminator. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 2, pp. 1331–1334. IEEE, New York (1997)
20. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. *IEEE Trans. Speech Audio Process.* **10**(5), 293–302 (2002)
21. Wang, D., Brown, G.J.: Computational Auditory Scene Analysis. Wiley, Hoboken, NJ (2006)
22. Wold, E., Blum, T., Keislar, D., Wheaton, J.: Content-based classification, search, and retrieval of audio. *IEEE MultiMedia* **3**(3), 27–36 (1996)
23. Yu, D., Deng, L.: Automatic Speech Recognition: A Deep Learning Approach. Signals and Communication Technology. Springer, London (2014)
24. Zhang, T., Kuo, C.C.J.: Audio content analysis for online audiovisual data segmentation and classification. *IEEE Trans. Speech Audio Process.* **9**(4), 441–457 (2001)

Chapter 2

The Machine Learning Approach for Analysis of Sound Scenes and Events

Toni Heittola, Emre Çakır, and Tuomas Virtanen

Abstract This chapter explains the basic concepts in computational methods used for analysis of sound scenes and events. Even though the analysis tasks in many applications seem different, the underlying computational methods are typically based on the same principles. We explain the commonalities between analysis tasks such as sound event detection, sound scene classification, or audio tagging. We focus on the machine learning approach, where the sound categories (i.e., classes) to be analyzed are defined in advance. We explain the typical components of an analysis system, including signal pre-processing, feature extraction, and pattern classification. We also present an example system based on multi-label deep neural networks, which has been found to be applicable in many analysis tasks discussed in this book. Finally, we explain the whole processing chain that involves developing computational audio analysis systems.

Keywords Audio analysis system • Sound classification • Sound event detection • Audio tagging • Machine learning • Supervised learning • Neural networks • Single-label classification • Multi-label classification • Acoustic feature extraction • System development process

2.1 Introduction

In each application related to computational sound scene and event analysis, the systems doing the computation need to solve very different types of tasks, for example, automatically detecting a baby crying, labeling videos with some predefined tags, or detecting whether a mobile phone is indoors or outdoors.

T. Heittola (✉) • E. Çakır
Tampere University of Technology, P.O. Box 527, FI-33101 Tampere, Finland
e-mail: toni.heittola@tut.fi; emre.cakir@tut.fi; tuomas.virtanen@tut.fi

T. Virtanen
Laboratory of Signal Processing, Tampere University of Technology, Tampere, Finland

Even though the tasks appear to be very different, the computational methods used actually share several similarities, and follow the same kind of processing architecture.

Sounds present in natural environments have substantial diversity, and, for example, semantically similar sound events have generally different acoustic characteristics. Because of this, the manual development of computational indicators of sound scene or event presence is viable only in very simple cases, e.g., a gunshot might be possible to detect simply based on loudness of the sound event. However, in many practical computational analysis systems the target sounds have more diverse characteristics and the system is required to detect more than one type of sounds. Depending on the target application, the number of classes may vary quite much between different analysis systems. In the simplest case, a detection system uses only two classes of sounds: the target sound class vs. all the other sounds. Theoretically there is no upper limit for the number of classes, but in practice it is limited by the available data that is used to develop systems, the accuracy that can be reached, and computational and memory requirements. In the scenario where there are multiple target classes, systems can also be categorized depending on whether they are able to detect only one event at a time, or multiple temporally overlapping events (which are often present in natural environments). Analyzing a large variety of sounds requires calculating a larger number of parameters from sound signals, and using automatic methods like *machine learning* [3, 9, 13, 22] to differentiate between various types of sounds.

Most of the computational analysis systems dealing with realistic sounds are based on the *supervised machine learning* approach, where the system is trained using labeled examples of sounds from each of target sound type [3, p. 3]. The supervised learning approach requires that there is a set of possible scene (e.g., street, home, office) or event (e.g., car passing by, footsteps, dog barking) categories, *classes*, defined by the system developer, and that there is sufficient amount of labeled examples available to train the system. Other machine learning techniques such as *unsupervised learning* [9, p. 17] and *semi-supervised learning* [9, p. 18] are applicable, however, in this book we largely concentrate on the supervised learning approaches, as they are the most frequently studied and used for the analysis of sound scenes and events.

This chapter gives a general overview of the supervised machine learning approach to analysis of sound scenes and events. Section 2.2 starts by presenting the overview of audio analysis systems and introducing the main processing blocks on such systems. Section 2.3 deals with the acquisition of learning examples, and Sect. 2.4 introduces the processing pipeline to transform the audio signals into a compact representations suitable for machine learning. Basics of supervised learning, including acoustic models, generalization properties, and recognition process are discussed in Sect. 2.5, followed in Sect. 2.6 by an example approach based on neural networks. Lastly, Sect. 2.7 presents the development process of the audio analysis systems from problem definition to functional application.

2.2 Analysis Systems Overview

Analysis systems can be categorized into two types depending on whether or not they output temporal information of sounds analyzed. Systems which output information about the temporal activity of target sound classes are said to perform *detection*. In this case, various temporal resolutions can be used, depending on the requirements of the application. Detection can be performed for one or more sound classes at a time. In the case where temporal information is not outputted, a system only indicates whether the sound classes to be analyzed are present in the item subject to analysis (e.g., a video recording, an audio file). A system is said to do *classification* when it can output only one of the possible classes for an item to be analyzed, and it is said to do *tagging*, when it can output more than one class simultaneously for an item to be analyzed. In the machine learning terminology, tagging would be equivalent to multi-label classification. Different analysis systems types are illustrated in Fig. 2.1.

Figure 2.2 presents the block diagram of a typical computational sound scene or event analysis system based on machine learning. The system takes an audio signal as input, either in real-time, captured by a microphone, or offline, from an audio recording. The methods presented in this book assume discrete-time signals, obtained by using analog-to-digital converters. The *audio processing* block consists of different processing stages and outputs *acoustic features*, as the actual analysis of audio is rarely based on the audio signal itself, but rather on the compact signal representation with features. The purpose of the feature extraction is to obtain information sufficient for detecting or classifying the target sounds, making the subsequent modeling stage computationally cheaper and also easier to achieve with limited amount of development material.

At the development stage, the obtained acoustic features are used together with *reference annotations* of the audio training examples, to learn models for the sound classes of interest. Annotations contain information about the presence of target sound classes in the training data, and are used as a reference information to automatically learn a mapping between acoustic features and class labels. The mapping is represented by *acoustic models*. At the usage stage, the learned acoustic

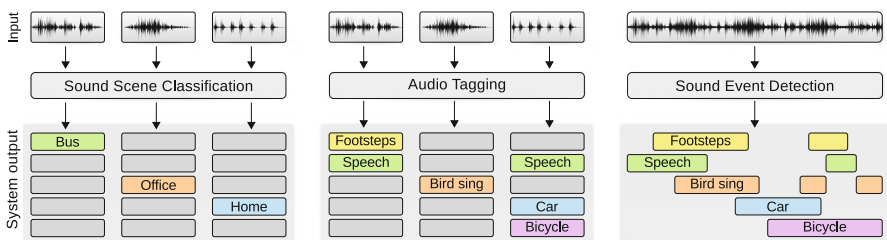


Fig. 2.1 System input and output characteristics for three analysis systems: sound scene classification, audio tagging, and sound event detection

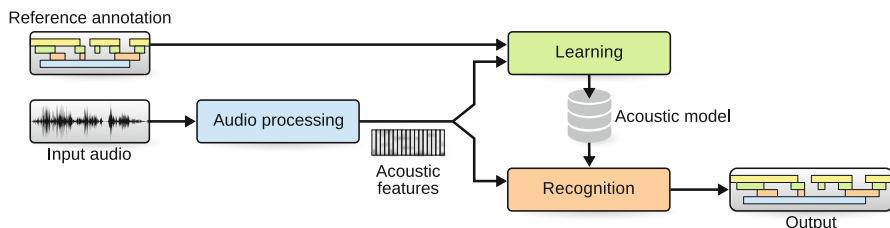


Fig. 2.2 Basic structure of an audio analysis system

models are used to do recognition (detection or classification), which predicts labels for the input audio. The recognition stage may also involve temporal models and post-processing of labels.

2.3 Data Acquisition

Machine learning approaches rely on data to learn parameters of the acoustic models, and to evaluate the performance of the learned acoustic models. The data includes both audio material and reference metadata associated with it (e.g., class labels). Data acquisition is an important stage of the development, as the performance of the developed system is highly dependent on the data used to develop it. As implementations of machine learning approaches are typically available, obtaining suitable training and development material is often one of the most time-consuming parts of the development cycle.

The defined target application dictates the type of acoustic data, recording conditions it is collected in, and type of metadata required. Essentially, the aim is to collect as realistic as possible acoustic signals in conditions which are as close as possible to the intended target application. Metadata should include a ground truth information which is often manually annotated during the data collection. Collected data should have sufficient amount of representative examples of all sound classes necessary for the target application to enable the acoustic models to *generalize* well [13, p. 107]. For preliminary feasibility studies, smaller datasets, containing only most typical examples can be collected. This type of dataset should not be used for the final system evaluation though, as there is higher danger that the acoustic models learned based on the dataset do not *generalize* well, and the system is optimized particularly for this small dataset. This section gives a brief overview of factors affecting the selection of the material that should be used, and discusses shortly potential ways to obtain material for development. Available data sources are discussed in more detail in Chap. 6 of this book.

2.3.1 Source Audio

The optimal performance of methods based on supervised classification is achieved when the material used to train and develop the system matches with the actual material encountered at the usage stage. Realistic sound sources commonly have internal variations in sound producing mechanism which can be heard as differences in the sound they produce. In classification tasks targeting such sounds, these variations cause intra-class variability which should be taken into account when collecting training material. The amount of examples required to sufficiently capture this variability depends highly on the degree of intra-class variability as well as similarity of target sound classes. As a general rule of thumb, easy classification cases may require tens of sound examples whereas more challenging scenarios can require easily hundreds or thousands of examples to capture intra-class variability sufficiently.

Depending on the application, there can be also variability in the sound signal caused by, e.g., characteristics of acoustic environment (e.g., size of room, type of reflective surfaces), relative placement of the source and the microphone, the used capture device, and interfering noise sources. In the ideal case, the above factors should be matched between actual usage stage and the training material to ensure optimal performance. However, in typical realistic audio analysis scenarios many of these variabilities cannot be fully controlled, leading to some level of mismatch between the material used to train and develop the system and the material encountered in the usage stage, and eventually to poor performance. These variations can be taken into account in the learning stage by making sure that the training material contains a representative set of signals captured under different conditions [20, p. 116]. This technique is called *multi-condition* or mixed condition training.

Most of the factors causing the variability (acoustic space, relative placement of the source and microphone, and capturing device) are reflected in the overall acoustic characteristics of the captured sound signal, called *impulse response* or in specific cases room impulse response [20, p. 206]. Different impulse responses can be artificially added to the signals, essentially easing the data collection process when using multi-condition training [41]. An effective strategy to achieve this is to obtain recordings of the target source with as little external effects as possible, and then simulate the effect of various impulse responses by convolving the signal with a collection of measured impulse responses from real acoustic environments. If measured impulse responses are not available, room simulation techniques can be used to generate room impulse responses for different type of acoustic environments [42, p. 191]. Similar strategies can be applied to interfering noise sources. If the noise source is known and stationary at the usage stage, the training material is relatively easy to collect under similar conditions. In the case where there are different types of noise sources at varying positions related to the microphone, the best resort is to use multi-condition training, i.e., include as many expected noise sources in the training material as possible. If it is feasible to obtain recordings of

target sound sources without any interfering noise and recordings with the noise sources alone, the best strategy is to simulate noisy source signals by artificially mixing these two types of recordings with various signal-to-noise ratios (SNR). This typically allows producing larger quantities of relevant training material in comparison to directly recording noisy material. On the other hand, the amount and diversity of the available recordings will influence and perhaps limit the variability of the produced material.

In order to start development quickly, source material can be obtained from external sources such as sound libraries (see Chap. 6 for more information). However, the availability of datasets that are collected for the development of supervised classification methods is limited, and the above discussed factors in the available datasets cannot be controlled properly. Therefore many audio analysis applications require collection of additional material for the development in order to achieve the best performance.

2.3.2 Reference Annotations

Supervised learning approaches that are discussed in this book require reference annotations, which indicate in which parts of the source audio each of the source classes is present. Depending on how the annotations are acquired, the annotations can be in different forms. Ideally the annotations will contain temporal information about each class, i.e., when a sound corresponding to the target class starts and when it ends. In practice, accurate temporal information can be difficult to obtain. Often the annotations are segment-level, i.e., each annotation indicates which classes are present in a segment of audio, but there is no temporal information about the class activities [19]. These two annotation types are illustrated in Fig. 2.3.

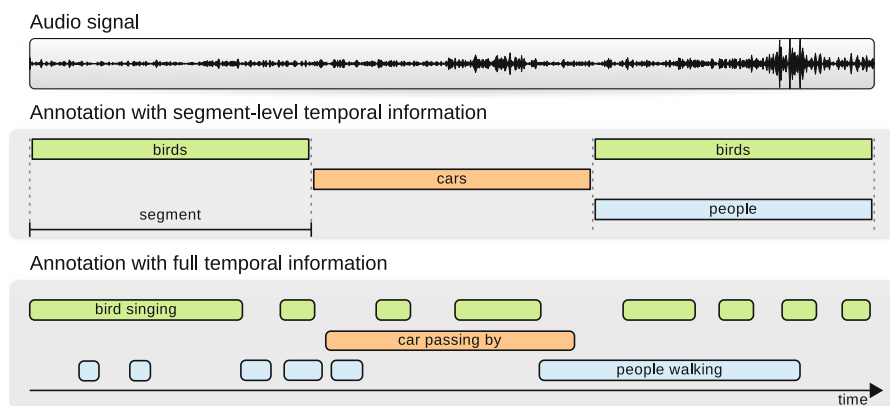


Fig. 2.3 Annotation with segment-level temporal information and with full temporal information

Reference annotations can be obtained in various ways. The most generic way to obtain annotations is to do it manually, i.e., by having persons listening to the audio to be used and indicating the activities of each class. This is a very time-consuming process, and annotating a piece of audio accurately takes easily much more time than the length of the audio. On the other hand, human annotation is often the only option to obtain annotations for certain types of sound classes. Human annotation is also the most generic approach since human annotators can be trained to annotate various types of classes. In addition to being slow, human annotation can be inaccurate at least if the material to be annotated is very noisy. Human annotations can also be subjective, which should be taken into account when the annotations are used as a reference when measuring the performance of developed methods. Details on producing annotations and validating their quality can be found in Chap. 6.

Sometimes it is possible to use other sensors to acquire the reference annotations. For example, the activity of a machine can be measured based on the electric power used by the machine or presence of moving cars can be detected from a video signal. This type of extra information may be available only during the development stage, while in the actual usage scenario the system must rely only on audio capture.

When training material is obtained from sample libraries or databases, the database often contains information about its content that can be used to obtain the annotations. However, the descriptions of the database may not match one-to-one the target classes and may therefore require some screening to identify and exclude possible mismatches.

2.4 Audio Processing

Audio is prepared and processed for machine learning algorithms in the audio processing phase of the overall system design. This phase consists of two stages: *pre-processing*, in which the audio signal is processed to reduce the effect of noise or to emphasize the target sounds, and *acoustic feature extraction*, in which the audio signal is transformed into a compact representation.

2.4.1 Pre-processing

Pre-processing is applied to the audio signal before acoustic feature extraction if needed. The main role of this stage is to enhance certain characteristics of the incoming signal in order to maximize audio analysis performance in the later phases of the analysis system. This is achieved by reducing the effects of noise or by emphasizing the target sounds in the signal.

If the audio data is collected from various sources, it is most likely captured in non-uniform recording settings, with variations in the amount of captured audio channel, and used sampling frequency. These variations can be addressed by converting the audio signal into uniform format by *down-mixing* it into fixed number of channels and *re-sampling* it into fixed sampling frequency.

Knowledge about the recording conditions and characteristics of target sounds can be utilized in the pre-processing stage to enhance the signal. The energy of audio signal is concentrated on lower frequencies; however, for sound recognition higher frequencies contain also important information. This issue can be addressed by *pre-emphasis*—emphasizing high frequencies before feature extraction. In the case of noisy environments, *noise suppression* techniques can be used to reduce interference of environmental noise to the audio analysis [31], while interference of overlapping sounds can be minimized by using *sound source separation* techniques [16].

2.4.2 Feature Extraction

The audio analysis is commonly based on acoustic features extracted from audio signal to represent the audio in a compact and non-redundant way. For recognition algorithms, the necessary property of the acoustic features is low variability among features extracted from examples assigned to the same class, and at the same time high variability allowing distinction between features extracted from example assigned to different classes [12, p. 107]. The feature representations fulfilling this property usually make the learning problem easier. A compact feature representation also requires less amount of memory and computational power than direct use of audio signal in the analysis.

The role of feature extraction is to transform the signal into a representation which maximizes the sound recognition performance of the analysis system. The acoustic features provide a numerical representation of the signal content relevant for machine learning, characterizing the signal with values which have connection to its physical properties, for example, signal energy, its distribution in frequency, and change over time. The processing pipeline in feature extraction is similar for many types of acoustic features used in analysis and consists of *frame blocking*, *windowing*, *spectrum* calculation, and subsequent analysis, as illustrated in Fig. 2.4.

Digital audio signals are discretized in terms of both amplitude and time when captured. For audio analysis, a significant amount of information is contained in relative distribution of energy in frequency, suggesting use of frequency domain features or time-frequency representations. The most common transformation used for audio signals is the discrete Fourier transform (DFT), which represents the signal with a superposition of sinusoidal base functions, each base being characterized by a magnitude and phase [25]. Examples of other transformation methods used for audio signals are constant-Q transform (CQT) [4] and discrete wavelet transform (DWT) [35].

Audio signals are generally non-stationary as the signal statistics (i.e., magnitudes of the frequency components) change rapidly over time. Because of this, the feature extraction utilizes the short-time processing approach, where the analysis is done periodically in short-time segments referred to as *analysis frames*, to capture the signal in quasi-stationary state. In *frame blocking* the audio signal is sliced into fixed length analysis frames, shifted with a fixed timestep. Typical analysis frame

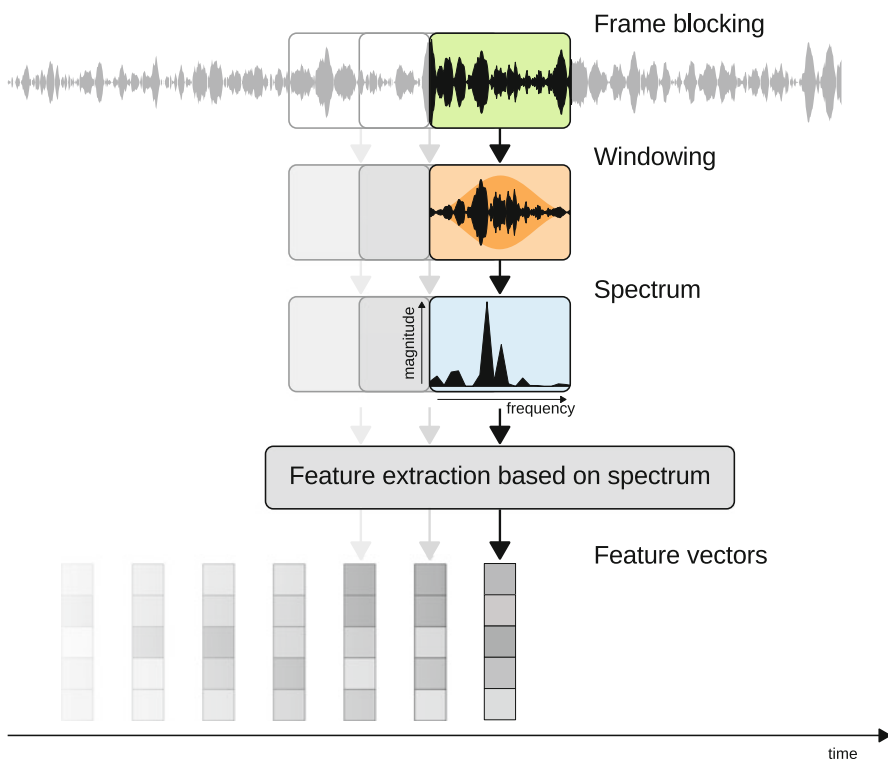


Fig. 2.4 The processing pipeline of feature extraction

sizes are between 20 and 60 ms, and the frame shift is typically selected so that the consecutive frames are overlapping at least 50%. The analysis frames are smoothed with a *windowing function* to avoid abrupt changes at the frame boundaries that can cause distortions in the spectrum. The windowed frame is then transformed into spectrum for further feature extraction.

The most common acoustic features used to represent spectral content of audio signals are mel-band energies and mel-frequency cepstral coefficients (MFCCs) [7]. Their design is based on the observation that human auditory perception focuses only on magnitudes of frequency components. The perception of these magnitudes is highly non-linear, and, in addition, perception of frequencies is also non-linear. Following perception, these acoustic feature extraction techniques use non-linear representation for magnitudes (power spectra and logarithm) and non-linear frequency scaling (mel-frequency scaling). The non-linear frequency scaling is implemented using filter banks which integrate the spectrum at non-linearly spaced frequency ranges, with narrow band-pass filters at low frequencies and with larger bandwidth at higher frequencies.

Mel-band energies and MFCCs provide a compact and smooth representation of the local spectrum, but neglect temporal changes in the spectrum over time, which

are also required for the recognition of environmental sounds. Temporal information can be included by using delta features, which represent the local slope of the extracted feature values within a predefined time window. Another way to capture the temporal aspect of the features is to stack feature vectors of neighboring frames (e.g., five on each side of the current frame) into a new feature vector.

The feature design and parameters used in the extraction commonly rely on prior knowledge and assumptions about the content of the acoustic signal, which in some analysis tasks can lead to sub-optimal performance. For example, selected length of the analysis frame or number of mel-bands might be optimal only for a subset of sound classes involved in the analysis. Unsupervised feature learning can be used to learn better fitted feature representations for specific tasks [29, 39]. The feature learning can be also incorporated into the overall learning process through end-to-end learning, thus avoiding explicit learning of the feature representation. In this approach, the correspondence of the input signal (usually raw audio signal or spectrogram) and desired recognition output is learned directly [8, 14, 17, 40].

2.5 Supervised Learning and Recognition

After the data acquisition and feature extraction steps, acoustic features and reference annotations for each audio signal are available. The next step is to *learn* a mapping between these features and class labels for sound classes, where the labels are determined from the reference annotations. This is based on a computational algorithm that can analyze and learn the similarities/differences between acoustic features and the class labels for various sound classes. The learned acoustic model is then used to assign a class label for acoustic features without reference annotations in the usage stage. The study of developing such algorithms is called *supervised learning*.

In supervised learning, we are given a set of input–target output pairs, and the aim is to learn a general model that maps the inputs to target outputs. In the case of classification of sound classes, we have acoustic features \mathbf{o}_t extracted from $t = 1, 2, \dots, T$ analysis frames and the reference annotations for each sound signal to be analyzed. Depending on the sound classification task at hand, there are several ways to define the input and the target output for the model (more details in Sect. 2.6). In this chapter, we define the input as $\mathbf{o}_t \in \mathbb{R}^F$, acoustic features extracted from a single analysis frame, where F is the number of features. The target output $\mathbf{y}_t \in \mathbb{R}^C$ is a binary vector which includes the annotation of present sound classes in the analysis frame among C predefined class labels. If, according to the reference annotations, the class with the c th label is present in the analysis frame t , then $y_{c,t}$ is set to 1 and 0 vice versa. Therefore, the acoustic features \mathbf{o}_t and the target outputs \mathbf{y}_t for each analysis frame correspond to a single input–target output pair, and each pair is called a *training example* for the model.

As illustrated in Fig. 2.5, the acoustic model is trained to learn the relationship between \mathbf{o}_t , the input feature vectors, and \mathbf{y}_t , the target outputs obtained from

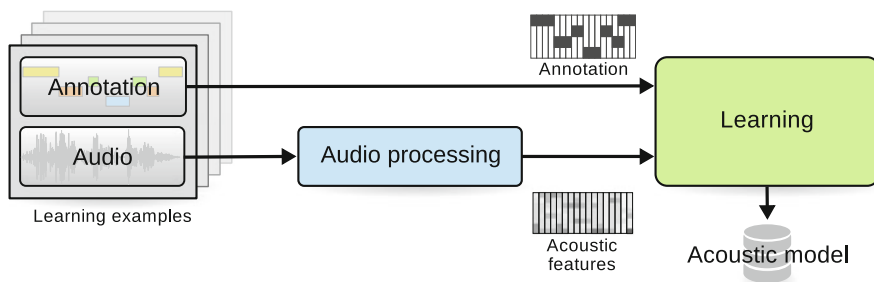


Fig. 2.5 Overview of supervised learning process for audio analysis

reference annotations. When the target output is chosen in the range $[0, 1]$, the model's estimated output $\hat{y}_{c,t} \in [0, 1]$ is expected to be either (a): close to 0 when class with c th label is *not* present, or (b): close to 1 when class with c th label is present. Therefore \hat{y}_t can be regarded as the *class presence probabilities*.

The type of the classification problem is an important factor in designing the model. If there can be at most one label present at a given frame, the task is regarded as *single-label* classification. Scene classification and sound event classification tasks are most often single label. The task of classifying multiple labels simultaneously present in a given frame is called *multi-label* classification. Sound event detection in real-life environments may belong to this category, since multiple sound events can occur simultaneously in daily life.

2.5.1 Learning

The learning process is about searching for the optimal model that would separate the examples from different classes on a given feature space. In Fig. 2.6, we illustrate a simple learning task involving examples with two features $\{o_1, o_2\}$ from two different classes marked with blue triangles and orange circles. The curved line that divides the examples from different classes is called the *decision boundary*. It is composed of data points that the model estimates to be equally likely belong to one of the two classes. In the given figure, it can be observed that some of the examples end up in the wrong side of the decision boundary, so our model can be deemed imperfect. The performance of the model is calculated through a *loss* (can be also called error or cost) function that calculates the difference between the target and estimated outputs for the training examples, and the model is updated in order to decrease the loss through various optimization techniques. For instance, we can initialize our model parameters so that the decision boundary is a flat line roughly dividing the examples from two classes. Then, we can iteratively update the model parameters by minimizing the mean squared error between the target outputs and estimated outputs based on the decision boundary.

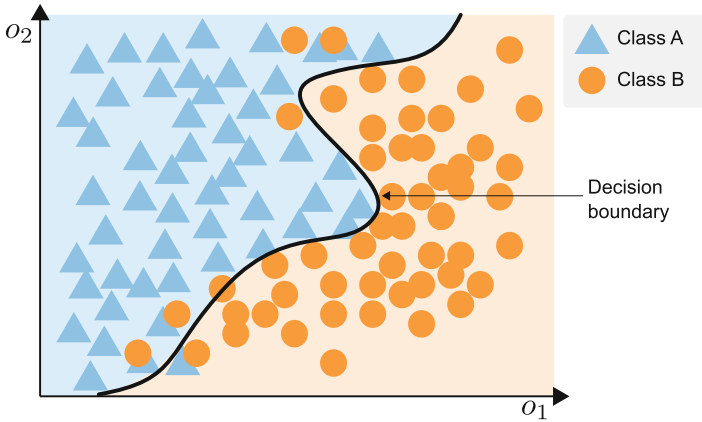


Fig. 2.6 Examples from two different classes and the decision boundary estimated by the learned model

Supervised learning methods are often grouped into two main categories: *generative* and *discriminative* learning methods. In generative learning, the aim is to model the joint distribution $p(x, y)$ for each class separately and then use Bayes' rule to find maximum posterior $p(y|x)$, i.e., from which class a given input x is most likely to be generated. Some of the established generative classifiers include Gaussian mixture models (GMM), hidden Markov models (HMM), and naive Bayes classifiers. On the other hand, in discriminative learning, the aim is to model the boundaries between the classes rather than the classes themselves and find a direct mapping between the input and the target output pairs [23]. Neural networks, decision trees, and support vector machines are some of the established discriminative learning methods. When it comes to classification of sound classes, discriminative learning has recently been the widely chosen method [5, 11, 26, 28]. This is due to the fact that high intra-class variability among the samples makes it hard to model the individual class accurately and also there is only little benefit for classification in doing so. For example, if our task is to classify an audio recording as either a cat meow or a dog bark, there is no need to model the cat meow and dog bark sounds individually, as long as we can distinguish these two classes from each other.

2.5.2 Generalization

Supervised learning methods aim to learn a model that can map the inputs to their target outputs for a given set of training examples. For the usage stage, the learned model is used to estimate the outputs for a different set of examples, which have not been used during learning stage. These examples are often called *test* (or *unseen*)

examples. The underlying assumption in the usage stage is that the test examples from a sound class have similar inputs compared to the inputs for the training examples for the same class. Therefore, if the mapping between the input acoustic features and the class label has been learned by the model during the learning stage, then the learned model should be able to estimate the correct outputs for test examples. However, in practice, the performance of the learned model may differ between the training examples and the test examples. In machine learning, the ability to perform well on unseen examples is called *generalization* [13, p. 107].

For sound classification tasks, there are several factors that make it challenging to reach a good degree of generalization. Due to high levels of environmental noise and multi-source nature of the recordings, there can be a large amount of variance among the examples from the same class, i.e., intra-class variability. Besides, class labels are often broadly defined to include a wide range of sound sources with high variation in their acoustic characteristics, such as *door bell* or *bird singing* (see Chap. 7 on taxonomy of sound events).

Modern supervised learning techniques, such as deep learning, are known for their ability to express highly non-linear relationships between the input and the output, given the high depth and large number of parameters [13, p. 163]. However, high expressional capability may also cause *overfitting* [15]. Overfitting is the term used when the difference between loss for training and test examples is large, i.e., the trained model can effectively model the examples for the training set but fails to generalize for the examples in the test set. A learned model with high accuracy in training examples and low accuracy in test examples may indicate that the model has learned the peculiarities of the training examples for better performance on the training set rather than to learn the general acoustic characteristics of the sound classes. Therefore, a sufficiently large number of examples that can provide the general characteristics of the classes and reflect the intra-class variability are required in the training set. Overfitting can also be reduced by using simpler approximation functions and regularization techniques such as L1/L2 weight regularization [24], dropout [32], and batch normalization [18]. On the other hand, the model should be complex enough to provide a good representation of the classes and low loss on the training set to avoid *underfitting* [37]. To summarize, learning is about finding the optimal model on the fine line between overfitting and underfitting.

2.5.3 Recognition

After an acoustic model for classification is obtained through the learning stage, the model is ready to be used in an actual usage scenario. An overview of the recognition process is shown in Fig. 2.7. First, acoustic features \mathbf{o}_t from the test examples are extracted. Then, frame-level class presence probabilities $\hat{\mathbf{y}}_t$ for the acoustic features are obtained through the learned model. Frame-level class presence probabilities $\hat{\mathbf{y}}_t$ can be obtained both from acoustic features \mathbf{o}_t in each timestep, or one can use a memory-based model such as recurrent neural networks to calculate $\hat{\mathbf{y}}_t$ from

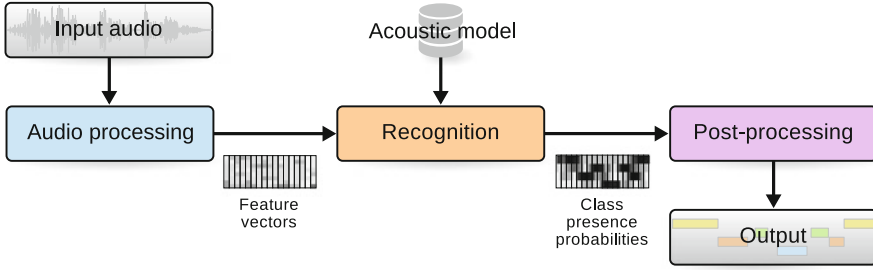


Fig. 2.7 Basic structure of recognition process

$\mathbf{o}_{(t-T_M):t}$, where T_M represents the memory capacity in timesteps. After obtaining $\hat{\mathbf{y}}_t$, there are several ways to obtain item-level *binary estimates* $\mathbf{z} \in \mathbb{R}^C$, which is a binary vector with only the assigned label elements as 1 and the rest 0, depending on the type of the classification task and the item to be analyzed.

Classification For a single-label sound classification task, the item to be analyzed (e.g., audio file or video recording) consists of multiple short analysis frames. In order to combine the class presence probabilities of multiple analysis frames in a single classification output, one can assign each frame the label with highest class presence probability. This way, one would obtain 1-hot *frame-level binary estimates* $\mathbf{z}_t \in \mathbb{R}^C$. Item-level binary estimates \mathbf{z} can be obtained, e.g., by performing a majority voting over the frame-level binary estimates of all the frames for the item, i.e., the item would be assigned the label with the highest number of occurrences among the estimated labels. Another way to obtain \mathbf{z} would be to sum $\hat{\mathbf{y}}_t \in \mathbb{R}^C$ class-wise among the frames of the item, and then assign the item the label with highest combined probability.

For a multi-label sound classification task, such as tagging, the number of present sound classes in each item is most often unknown, so a similar majority voting approach cannot be applied. In that case, frame-level class presence probabilities $\hat{\mathbf{y}}_t$ can be converted to item-level class presence probabilities $\hat{\mathbf{y}}$, e.g., by taking the average or the maximum $\hat{\mathbf{y}}_t$ for each class among all the frames of the item. Taking the maximum $\hat{\mathbf{y}}_t$ among all the frames would help to correctly classify the classes with rare activity (and therefore low average presence probability over the frames). On the other hand, taking the average $\hat{\mathbf{y}}_t$ would be useful for the cases when a class is mistakenly assigned a high presence probability in a small portion of frames (since the average probability over the frames would be low in this case).

Then, binary estimates \mathbf{z} for the item can be obtained by converting $\hat{\mathbf{y}}$ into a binary vector over a certain binarization rule. A simple binarization rule would be thresholding over a constant σ subject to $0 < \sigma < 1$.

Detection and Temporal Post-processing In order to obtain the temporal activity information of the sound classes in the usage stage, the acoustic features $\{\mathbf{o}_r\}_{r=1}^T$ are presented to the acoustic model in a time sequential form. The features are extracted

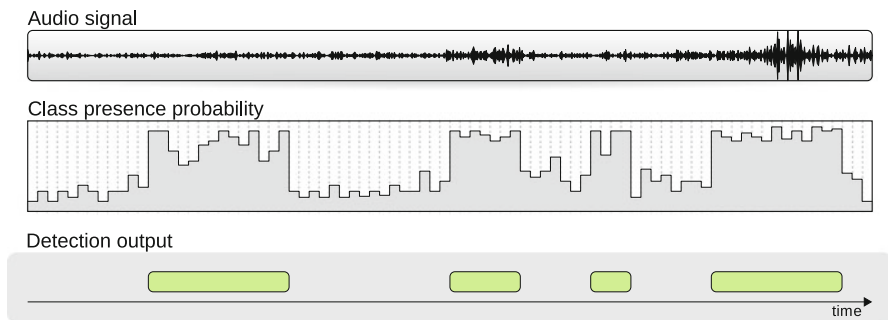


Fig. 2.8 Temporal activity information for each class can be obtained from the class presence probabilities for consecutive analysis frames

from the consecutive analysis frames of the item to be analyzed and frame-level class presence probabilities are obtained through the learned model. Detection of the temporal activity for a single class from frame-level class presence probabilities is visualized in Fig. 2.8.

The simplest way of obtaining discrete decisions about source activities from frame-level class presence probabilities \hat{y}_t is to first convert \hat{y}_t to frame-level binary estimates z_t over a certain binarization rule, such as the thresholding that was presented above. Having a frame-level binary estimate $z_{c,t}$ for class c in consecutive frames allows us to estimate the onset and offset information for this class. This way, the temporal position of each sound class can be *detected* among the audio signal.

When spectral domain acoustic features are used, the typical values selected for the analysis frame length are quite small (often ranging from 20 to 50 ms). On the other hand, the duration of an individual sound event is most often longer than the analysis frame length, and a single event can span several consecutive frames. For a given acoustic model, this may result in a correlation between classification outputs for consecutive analysis frames. In order to make use of this correlation in the detection tasks, temporal post-processing techniques can be applied over either frame-level class presence probabilities \hat{y}_t or binary estimates z_t . There are several temporal post-processing techniques, and next, we will shortly describe two of them.

Sound signals may have short, intermittent periods which do not reflect the acoustic characteristics of the sound class that they have been labeled with. For instance, due to the overlap between the feature distributions over different sound classes, acoustic features for an analysis frame for a sound class may be very similar to the features from another class. Therefore, processing the audio signal in short, consecutive analysis frames may introduce some noise in the detection outputs. One simple way to filter this noise and smoothen the detection outputs is to use *median filtering*. For each frame, the post-processed frame-level binary estimate \tilde{z}_t is obtained by taking the median of the binary estimates in a window of frames [5]. The method is continued by sliding this window by one frame when every new frame is classified through the model.

Hidden Markov model (HMM) is an established generative learning method which can be used for temporal post-processing over class presence probabilities [10, 11]. HMM can be used for (a): smoothing $\hat{\mathbf{y}}_t$ using class presence probabilities from previous analysis frames, and (b): producing an estimate of a hidden state variable for each state which allows segmenting a sequence of features from consecutive frames to various sound classes, provided that HMM states are class-specific. More detailed information about HMMs can be found in Chap. 5.

2.6 An Example Approach Based on Neural Networks

This section introduces a basic deep neural network (DNN) [13, 30] based approach for general audio analysis. DNNs are discriminative classifiers that can model the highly non-linear relationships between the inputs and outputs, and which can be easily adapted to output multiple classes at a time (multi-label classification) [5, 26]. This is especially useful for real-life environmental audio, as sounds are very likely to overlap in time. For example, a recording in street environment may include sounds such as car horns, speech, car engines, and footsteps occurring at the same time. DNNs also enable good scalability. Depending on the computational resources available and performance requirements of the target application, the network size can be adjusted accordingly. With larger network sizes, DNNs can take advantage of large sets of examples in the learning process, thus covering a high variability of sounds. This usually leads to better generalization of the acoustic model and better overall performance. With smaller network sizes, the approach can meet the computational limits of the application without compromising the performance too much. DNN-based audio analysis systems have recently shown superior performance over more traditional machine learning approaches (e.g., GMM and support vector machines) given that there is sufficiently large amount of learning examples available [5, 26]. The presented basic system architecture is followed in many current state-of-the-art systems with various extensions, for example [1, 5, 11, 26]. The architecture is here differentiated for two target applications: audio tagging and sound event detection. Even though these applications may seem at first quite dissimilar, the system architectures for them are highly similar, allowing easy switching between applications during the research and development as will be explained. The basic system architecture is illustrated in Fig. 2.9. The system uses DNNs to classify input audio in analysis frames, and using the frame-wise classification results to get a system output matching the requirements of the target application. Collected data, audio signals, and associated reference annotations are divided into non-overlapping training and test sets.

Learning Stage In the learning stage, the training set is used to learn the acoustic model. Training examples consist of acoustic features extracted in short analysis frames from audio signals and target outputs defined for each frame based on reference annotations. Acoustic features are extracted in 40 ms analysis frames with

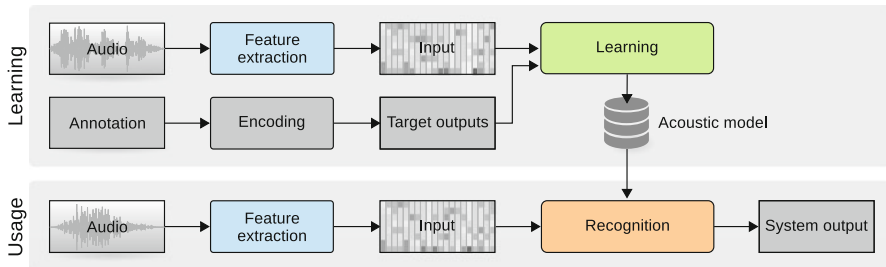


Fig. 2.9 Framework of the training and testing procedure for the example system. Grey processing blocks are adapted according to the target application for the system

50% overlap over the audio signal. For each analysis frame, the magnitude spectrum is calculated as the magnitude of the discrete Fourier transform (DFT) of the frame. The bins are accumulated into 40 mel bands using non-linear frequency scaling spanning from 0 to 22.05 kHz (assuming audio signal is sampled with 44.1 kHz) and logarithm is applied to get the acoustic features—the *log mel band energies* [33, 34]. To include a temporal aspect of the features into the final feature vector, *frame stacking* can be used: the acoustic features extracted from the current frame are concatenated with features from neighboring frames, e.g., the previous four frames, to create a single feature vector. The target output vectors for each analysis frame are obtained by binary encoding of the reference annotations. In this process, classes annotated to be active within the current analysis frame are marked with 1 and non-active classes are marked with 0. The target outputs for the training set examples will be used in training the acoustic model. A DNN acoustic model is used to learn a mapping between acoustic features and the target outputs for the training set examples [13, p. 163].

The DNN consists of multiple layers of inter-connected elements called *neurons*. Each neuron outputs an activation through a weighted sum of previous layer activations and a non-linear activation function. The first layer takes as input the acoustic features and the following layers take as input the previous layer activations. The class presence probabilities are obtained through the activations of the final layer. The network parameters (weights of the neurons) are learned through an iterative process where parameters are updated using an optimization algorithm (e.g., gradient descent) and a loss function (e.g., cross-entropy) [13, p. 171]. Part of the training examples are left out from the actual learning process, for *validation*, being used to evaluate the performance of the system between the learning iterations and to decide when the learning process should be stopped to avoid overfitting [13, p. 239]. A comprehensive description of DNNs can be found in [13] and a review of the DNN-based techniques can be found in [30]. After the network parameters are trained, the system is ready to be used for test examples.

Usage Stage The usage stage shares the acoustic feature extraction part (same parameters) with the learning stage. The same acoustic features are extracted from

the input audio, and the previously learned acoustic model is used to get the class presence probabilities for each analysis frame. The class presence probabilities are acquired by a single forward-pass through the learned network. Frame-wise class presence probabilities are then processed to obtain the output in correct format for the target application as discussed in next subsections.

2.6.1 Sound Classification

The previously presented system structure for audio analysis can be adapted for classification applications through specific use of training examples, output layer for the network, and post-processing of the frame-wise class presence probabilities. In the classification task, a segment of audio is classified into a single predefined class for *single-label classification*, or into multiple predefined classes for *multi-label classification*, depending on the target application. Audio analysis systems performing multi-label classification are also referred to as *audio tagging* systems. Illustrative examples of system inputs and outputs for these applications are shown in Fig. 2.1.

Single-Label Classification For single-label classification, the learning examples are audio segments with a single class annotated throughout. The annotations are encoded into target outputs which are used in the learning stage together with audio signals. In this case, classes are mutually exclusive. This condition is included into the neural network architecture by using output layer with *softmax* activation function, which will normalize outputted frame-level class presence probabilities to sum up to one [13, p. 78]. In the usage stage, the frame-level class presence probabilities within the classified item (e.g., audio segment) are first calculated. These probabilities can be used to get the overall classification output in two different ways: by doing classification at frame-level and combining results, or by combining frame-level information and doing final classification at item level. In the frame-level approach, classification is done first for each frame by assigning each frame the label with the highest class presence probability, and then majority voting is used among these frame-level results to get the final classification result. In the item-level approach, the frame-level class presence probabilities are summed up class-wise and the final classification is done by assigning the item the label with highest combined probability. This type of system architecture has been utilized for both acoustic scene classification [2, 27, 36] and sound event classification tasks [21, 28].

Multi-Label Classification For multi-label classification or audio tagging, the learning examples contain audio annotated similarly as for single-label classification, only this time multiple classes can be active at same time in the annotations. In this case, the neural network architecture is using an output layer with *sigmoid* activation, which will output class presence probabilities independently in the range (0, 1) [13, p. 65]. In the usage stage, the frame-level class presence

probabilities within the classified item (e.g., audio segment) are calculated and collected over the item. The final class-wise activity estimation is done, for example, based on the average frame-level class presence probability and binarization with a threshold σ . Since the average frame-level class presence probability is in the range $[0,1]$, an unbiased selection for σ would be 0.5 [5]. The threshold σ can be adjusted if there is any bias towards less false-alarms (false positives) or less misses (false negatives) in the usage stage. The same overall system architecture has been used in many published works [6, 28, 38]. A system architecture for multi-label sound classification is shown in Fig. 2.10, where highlighted blocks are modified compared to the basic architecture (see Fig. 2.9 for comparison).

2.6.2 Sound Event Detection

The basic system structure can also be adapted for detection applications. In the detection task, temporal activity is estimated along with actual class labels for the events. A system architecture for sound event detection is shown in Fig. 2.11. The highlighted blocks are the ones different compared to the basic architecture from Fig. 2.9.

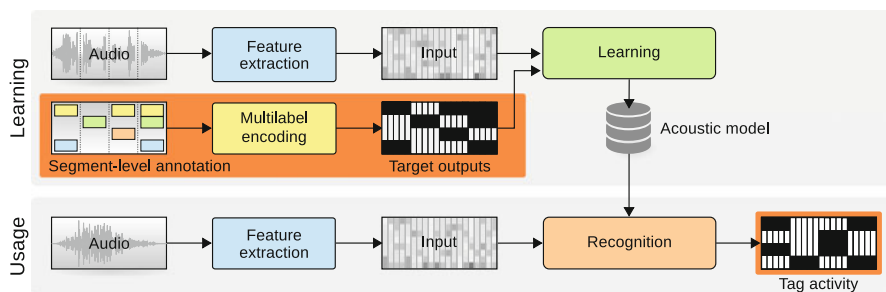


Fig. 2.10 Multi-label sound classification. Adapted blocks compared to the basic system architecture shown in Fig. 2.9 are highlighted

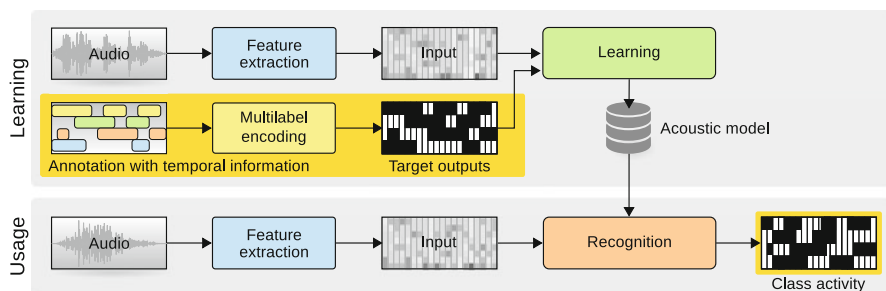


Fig. 2.11 Sound event detection. Adapted blocks compared to the basic system architecture shown in Fig. 2.9 are highlighted

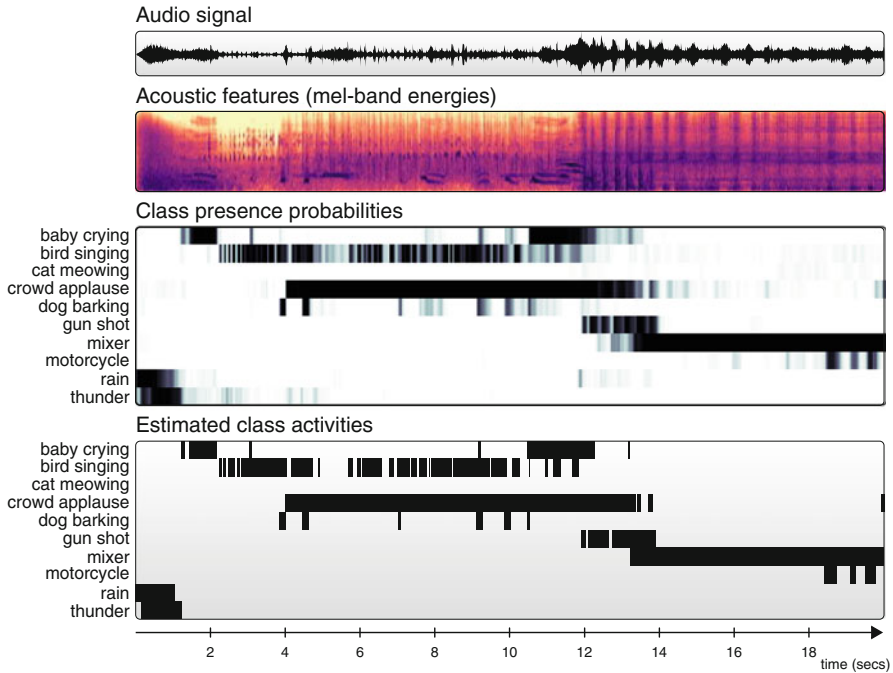


Fig. 2.12 Illustration of acoustic features, class presence probabilities, and estimated class activities for a multi-label sound event detection task

Essentially this is the same architecture as the multi-label classification one, the main difference being the temporal resolution of the acoustic modeling during the learning and usage stages. Annotation of sound events provides detailed temporal information about class presence at any given time, which is transformed into a frame-level binary activity indicator that serves as target for the network learning stage. In the usage stage, binarization of the class presence probabilities output by the network is done at frame-level, resulting in estimated class activities that are likely to be noisy. This procedure is illustrated in Fig. 2.12. Post-processing is commonly applied to this binarized output by imposing minimum lengths (e.g., 300 ms) for event instances and the gaps between them (e.g., 200 ms), to clean up activity estimates. This system architecture has been used successfully for sound event detection in recent years [1, 5, 11].

2.7 Development Process of Audio Analysis Systems

The previous sections presented the main building blocks of the audio analysis system, and introduced basic system architectures for various target applications. A majority of systems are nowadays based on machine learning methods, and to get

these type of systems to work at the optimal level, rigorous development process is required. The crucial steps required for successful system development are identified and briefly introduced in this section, in order to give a general overview of the development process.

In academia, the audio analysis systems are often developed from a research perspective. The aim of this process is to push the technology in this field further, including through academic publications. Once the technology reaches a sufficient level, measured either in analysis performance or user experience, the development focuses on refining the technology and making it work reliably in real use cases, and possibly building a real product around it. Often in this stage the development starts to shift towards industry either through joint projects between academia and industry, or by moving intellectual properties to industrial partners. As the development progresses, the aim of the development is to deploy a product into actual use (either into commercial or non-commercial market).

The development of a audio analysis system consists of two main phases: *technological research*, where research and development is done in a laboratory environment, and *product demonstration* with a prototype system in a real operating environment. This book concentrates mostly on topics related to the first phase. The technological research phase uses fundamental research from different fields, and applies them to the target application. Interesting supporting research fields for sound scene and event analysis are, for example, human perception and machine learning. The aim of technological research is to produce a *proof-of-concept* system that shows the capability and feasibility of the analysis system, with the system evaluated in laboratory environment with audio datasets. In the product demonstration phase, the proof-of-concept system is further developed into a *prototype* having all the key features planned for the final system. The prototype is tested in realistic use cases, and demonstrated with real users, while continuing to develop and refine the system for eventually being suitable for deployment into use.

2.7.1 Technological Research

Before entering the active research and development of the audio analysis system, one has to identify the target application for the system and main characteristics of this application, as these will dictate system design choices later on. By having the target application identified, the research problem becomes easier to define. Sometimes in academic research, one cannot identify a clear target application for the system, especially in such early stage of the research. In these cases it is still a good practice to envision a speculative application for the system to steer the research and development.

In the research problem definition, the analysis system type is identified (detection *vs.* classification), the used sound classes are defined, and amount of classes needed to be recognized at the same time is defined (e.g., classification *vs.* tagging). For example, if our target application is the recognition of the sound scene in

5 s intervals from a predefined set of scene classes, then the problem definition would be the classification of input signal into a single class within predefined set of 15 sound scene classes. Another example would have as target application to recognize certain target sound events in office environment (e.g., mouse click); in this case the problem definition would be the detection of occurrences of the one predefined sound event in a predefined environment. Based on the defined research problem, the system development moves into active research and development. In this phase, three main stages can be identified: data collection, system design and implementation, and system evaluation.

Data Collection The audio data and associated metadata is used to learn parameters of the acoustic models in the system design and implementation stage. Furthermore, the data is used to evaluate performance in the system evaluation stage. The defined target application dictates the type of acoustic data, recording conditions it is collected in and type of metadata required. In general, the aim should be to collect as realistic as possible acoustic signals in conditions which are as close as possible to the target application. Details of the data collection and the annotation process are discussed in Chap. 6.

System Design and Implementation The main goal of the system design and implementation stage is to create a proof-of-concept system which solves the defined problem. The stage starts with the design of the overall system architecture and implementation of the initial version of the system. This initial version usually contains basic approaches and it is used as comparison point (*baseline system*) when developing the system further. The actual system design is a gradual process, where different steps of the system are developed either independently or jointly, and integrated into the overall system. Main steps involved in the processing chain for system design are audio processing (containing, e.g., pre-processing and acoustic feature extraction), machine learning (containing, e.g., learning, and recognition or detection), and post-processing. To some extent, it is possible to isolate each step in the development and optimize the corresponding parameters of the system while keeping the other parts fixed. This allows to take into account the effect of each processing step on the overall system performance, and will enable identification of the error sources and their contribution to the overall performance. The system integration stage uses the knowledge acquired in the development process to maximize performance of the overall system. In particular cases, some steps can be designed, implemented, and evaluated independently outside the system for optimal performance: for example, a noise suppression method can be first evaluated using specific data and metrics, before including it into the developed system as a pre-processing step.

System Evaluation The evaluation of the system is based on the test data and reference annotations assigned to it, and using a metric related to the target application. Ideally the system should be evaluated in scenarios which match the target application as much as possible, to get a realistic performance estimation. During the development the evaluation is commonly done in steps by gradually

increasing how closely the scenario matches the target application to better isolate factors affecting the performance. At the beginning of the development, highly controlled laboratory scenarios are preferred, and as the development progresses evaluation switches to more realistic scenarios. An example of a highly controlled scenario is the *offline* scenario where pre-recorded audio data is read directly from audio files and the analysis can be done efficiently for large datasets repeatedly while the core algorithms are developed. Depending on the target application, the system should be evaluated also in *online* use case, where the system is capturing audio in real-time. For example, the same pre-recorded audio data used in offline case can be played back in a laboratory environment to verify that the system performs similarly as in offline case and then move to more realistic usage environment. The evaluation metrics are chosen to reflect performance that should be maximized or errors that should be minimized in the system development. In some cases, subjective evaluation of the system can be performed based on user opinions or user satisfaction with system output, avoiding the need for reference annotations. For objective evaluation, a part of the data is assigned to the *training* of the system and a part is assigned to the *testing* of the system's performance. For sound scene recognition systems the most commonly used metric is accuracy, a percentage of correctly classified items. For sound event detection systems the most commonly used metrics are F-score (balanced mean of precision and recall) and error rate (ER). They are both calculated based on correctly classified items and errors made by the system, but emphasize different characteristics of the systems, namely the ability of correctly detecting as much as possible of the content versus the ability of making as small amount of mistakes as possible. The details of evaluation metrics are discussed in Chap. 6. The performance of the system is compared against other solutions to the same problem, for example, state-of-the-art methods or well-established (*baseline*) approaches. The analysis of these alternative solutions and comparison against developed system is necessary to put the obtained evaluation scores into larger context while doing the research.

2.7.2 *Product Demonstrations*

Once the proof-of-concept system is ready, the development moves to the product demonstration phase. In this phase, the desired use cases are first identified, and acceptable error types and level of performance in these use cases are defined. These factors are closely related to the end users' requirements and perception on good performing system, and thus they should be considered as early as possible in the development of the product. The proof-of-concept system developed in the technological research phase is used as a starting point for the development targeting a prototype having all the key features planned for the final system. When the prototype is ready, the technology is validated by testing it in real operating environment with realistic use cases, and demonstrated with real users. User experience studies can be used in this stage to get quantified feedback, and these results can be further used to refine the overall system design.

In the actual development project, there are often setbacks which force the development to return back to the technological research phase and to reiterate stages within it. Testing the system with realistic use cases can reveal problems in the core system design which were not identified earlier in the development. For example, the system could have poor user experience due to a wrongly selected recognition approach which is producing results with too high latency, or the system could have low recognition performance because of low noise robustness. These type of problems are usually such fundamental design flaws that they have to be addressed in the technological research phase.

Once the prototype is validated and achieves sufficient recognition performance with good user experience, the system is developed into a final system which is demonstrated with a small group of real users in actual operating environments. Usually in this stage the core system is considered ready and development concentrates mainly on polishing the possible user interface and communication interfaces with other applications. After the successful demonstrations, the system can be deployed to the market with small scale pilots first, and finally in full scale.

2.7.3 Development Process

The previously introduced development stages often overlap, and are executed multiple times during the overall development process. An example of one possible development process is shown in Fig. 2.13. The figure shows main stages for both

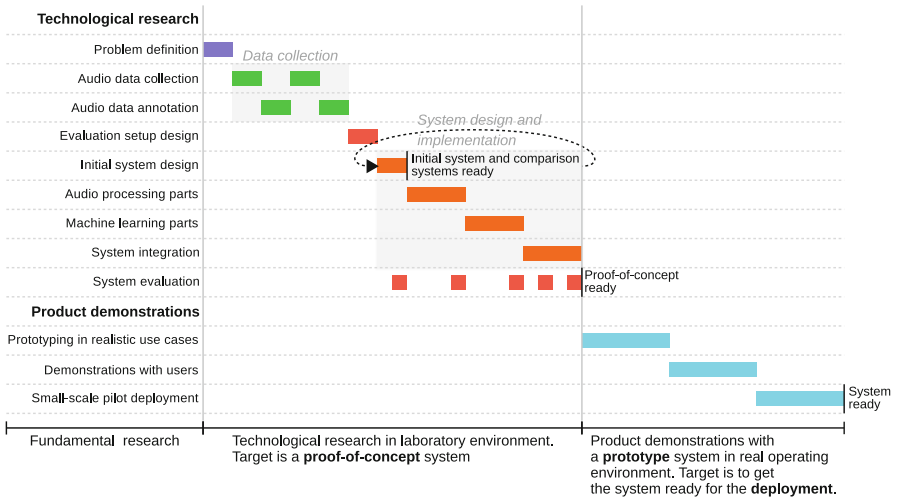


Fig. 2.13 Process graph for audio analysis system development. Fundamental research is happening outside this graph. The actual development is divided into two phases, technological research phase and product demonstration phase

the technological research and the product demonstration phases to give a comprehensive view for the whole process. However, as the book mainly concentrates on the technological research phase, in the following only the stages from technological research phase are fully explained.

The technological research phase starts from the problem definition stage, where the target application for the system is identified and the system characteristics are defined. After this, the actual active development starts by data collection and manual annotation of this collected data. The annotation stage is usually one of the most time-consuming parts of the development, and in some cases the amount of data is gradually increased as the annotation progresses during the development. It is also a good practice to interlace the data collection and annotation, to get some data ready for the development as early as possible. Once the dataset is complete, the evaluation setup is designed by selecting the evaluation metrics, defining the cross-validation setup, and selecting appropriate comparison systems. Before entering the full system development, the overall system architecture has to be designed. As each part of the system is developed in separate development stages but evaluated as a part of entire system, the components of the whole system have to be defined in general terms. The initial version of the entire system is implemented based on this design by using basic approaches, and usually is also used as a baseline system.

After the initial version of the system is ready, the main development can start. Individual parts of the system are designed, implemented, and evaluated in the order which follows the logical signal path through the system: first audio processing parts (pre-processing, acoustic feature extraction), followed by machine learning parts, and finally the post-processing parts. The evaluation results are used to guide the design choices within each part of the system. When the system performance reaches the desired level or saturates, the development moves to the next part. In the system integration stage, all parts of the system are optimized to get maximum overall performance. The end result of this stage is a complete proof-of-the-concept system which can be moved for further development to the product demonstration phase.

2.8 Conclusions

This chapter introduced the basic concepts in computational methods used for audio analysis, concentrating on supervised machine learning approaches. The focus is on classification and detection applications such as scene classification, audio tagging, and sound event detection. Systems for audio analysis have very similar architecture, with building blocks such as data acquisition, feature extraction, and learning often being identical between different systems. The learning process mirrors the selected target application in its association between labels and features, guiding the mapping between class labels and features performed during learning. An approach based on deep neural networks was presented, illustrating a very general system architecture that can be adapted for various sound classification and detection problems.

The last section of this chapter brought into discussion the larger picture of system development, from the definition of the problem all the way to the product developed for the market. System development is often a lengthy and iterative process involving stages of various difficulty and duration. Mid-way of this process is the link between academic research and industry, where the focus switches from proof-of-concept to the commercialization of a product. Industrial research concentrates on the prototyping and product demonstrations with the goal of refining and improving the user experience with the product. Specific solutions for this will be presented in more detail in Chap. 12.

References

1. Adavanne, S., Parascandolo, G., Pertila, P., Heittola, T., Virtanen, T.: Sound event detection in multichannel audio using spatial and harmonic features. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, pp. 6–10 (2016)
2. Bae, S.H., Choi, I., Kim, N.S.: Acoustic scene classification using parallel combination of LSTM and CNN. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, pp. 11–15 (2016)
3. Bishop, C.M.: *Pattern Recognition and Machine Learning*. Springer, New York (2006)
4. Brown, J.C.: Calculation of a constant q spectral transform. *J. Acoust. Soc. Am.* **89**(1), 425–434 (1991)
5. Çakır, E., Heittola, T., Huttunen, H., Virtanen, T.: Polyphonic sound event detection using multi label deep neural networks. In: *The International Joint Conference on Neural Networks 2015 (IJCNN 2015)* (2015)
6. Çakır, E., Heittola, T., Virtanen, T.: Domestic audio tagging with convolutional neural networks. Technical report, DCASE2016 Challenge (2016)
7. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech Signal Process.* **28**, 357–366 (1980)
8. Dieleman, S., Schrauwen, B.: End-to-end learning for music audio. In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6964–6968. IEEE (2014). doi:10.1109/ICASSP.2014.6854950
9. Du, K.L., Swamy, M.N.: *Neural Networks and Statistical Learning*. Springer Publishing Company, Incorporated, New York (2013)
10. Espi, M., Fujimoto, M., Kubo, Y., Nakatani, T.: Spectrogram patch based acoustic event detection and classification in speech overlapping conditions. In: *4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, pp. 117–121 (2014)
11. Espi, M., Fujimoto, M., Kinoshita, K., Nakatani, T.: Exploiting spectro-temporal locality in deep learning based acoustic event detection. *EURASIP J. Audio Speech Music Process.* **2015**(1), 26 (2015)
12. Gold, B., Morgan, N., Ellis, D.: *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*, 2nd edn. Wiley-Interscience, New York, NY (2011)
13. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press, Cambridge (2016)
14. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, ICML'14, vol. 14, pp. 1764–1772. JMLR Workshop and Conference Proceedings (2014)
15. Hawkins, D.M.: The problem of overfitting. *J. Chem. Inf. Comput. Sci.* **44**(1), 1–12 (2004)
16. Heittola, T., Mesaros, A., Virtanen, T., Gabbouj, M.: Supervised model training for overlapping sound events based on unsupervised source separation. In: *38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2013)*, pp. 8677–8681 (2013)

17. Hertel, L., Phan, H., Mertins, A.: Comparing time and frequency domain for audio event recognition using deep learning. In: Proceedings IEEE International Joint Conference on Neural Networks (IJCNN 2016), pp. 3407–3411 (2016)
18. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. CoRR abs/1502.03167 (2015)
19. Kumar, A., Raj, B.: Audio event detection using weakly labeled data. In: Proceedings of the 2016 ACM on Multimedia Conference, MM'16, pp. 1038–1047. ACM, New York (2016)
20. Li, J., Deng, L., Haeb-Umbach, R., Gong, Y.: Robust Automatic Speech Recognition — A Bridge to Practical Applications, 1st edn., 306 pp. Elsevier, Amsterdam (2015)
21. Lim, H., Kim, M.J., Kim, H.: Cross-acoustic transfer learning for sound event classification. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2504–2508 (2016)
22. Murphy, K.P.: Machine Learning: A Probabilistic Perspective. The MIT Press, Cambridge (2012)
23. Ng, A., Jordan, A.: On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. Adv. Neural Inf. Proces. Syst. **14**, 841 (2002)
24. Nowlan, S.J., Hinton, G.E.: Simplifying neural networks by soft weight-sharing. Neural Comput. **4**(4), 473–493 (1992)
25. Oppenheim, A.V., Schaffer, R.W., Buck, J.R.: Discrete-Time Signal Processing. Prentice Hall, Upper Saddle River, NJ (1999)
26. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6440–6444 (2016)
27. Petetin, Y., Laroche, C., Mayoue, A.: Deep neural networks for audio scene recognition. In: 23rd European Signal Processing Conference (EUSIPCO), pp. 125–129. IEEE, New York (2015)
28. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: IEEE International Workshop on Machine Learning for Signal Processing (2015)
29. Salomon, J., Bello, J.P.: Unsupervised feature learning for urban sound classification. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 171–175 (2015)
30. Schmidhuber, J.: Deep learning in neural networks: an overview. Neural Netw. **61**, 85–117 (2015)
31. Schröder, J., Moritz, N., Schädler, M.R., Cauchi, B., Adiloglu, K., Anemüller, J., Doclo, S., Kollmeier, B., Goetze, S.: On the use of spectro-temporal features for the IEEE AASP challenge 'detection and classification of acoustic scenes and events'. In: 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 1–4 (2013)
32. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
33. Stevens, S.S., Volkman, J.: The relation of pitch to frequency: a revised scale. Am. J. Psychol. **53**, 329–353 (1940)
34. Stevens, S.S., Volkman, J., Newman, E.B.: A scale for the measurement of the psychological magnitude pitch. J. Acoust. Soc. Am. **8**(3), 185–190 (1937)
35. Tzanetakis, G., Essl, G., Cook, P.R.: Audio analysis using the discrete wavelet transform. In: Proceedings of the WSES International Conference Acoustics and Music: Theory and Applications (AMTA 2001), Skiathos, pp. 318–323 (2001)
36. Valenti, M., Diment, A., Parascandolo, G., Squartini, S., Virtanen, T.: DCASE 2016 acoustic scene classification using convolutional neural networks. In: Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016), pp. 95–99 (2016)
37. Van der Aalst, W.M., Rubin, V., Verbeek, H., van Dongen, B.F., Kindler, E., Günther, C.W.: Process mining: a two-step approach to balance between underfitting and overfitting. Softw. Syst. Model. **9**(1), 87–111 (2010)

38. Xu, Y., Huang, Q., Wang, W., Jackson, P.J.B., Plumbley, M.D.: Fully DNN-based multi-label regression for audio tagging. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, pp. 105–109 (2016)
39. Xu, Y., Huang, Q., Wang, W., Foster, P., Sigtia, S., Jackson, P.J.B., Plumbley, M.D.: Unsupervised feature learning based on deep models for environmental audio tagging. *IEEE/ACM Trans. Audio Speech Lang. Process.* **25**, 1230–1241 (2017)
40. Yuji Tokozume, T.H.: Learning environmental sounds with end-to-end convolutional neural network. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2712–2725 (2017)
41. Zhao, X., Wang, Y., Wang, D.: Robust speaker identification in noisy and reverberant conditions. *IEEE/ACM Trans. Audio Speech Lang. Process.* **22**(4), 836–845 (2014)
42. Zölzer, U. (ed.): *Digital Audio Signal Processing*, 2nd edn. Wiley, New York (2008)

Chapter 3

Acoustics and Psychoacoustics of Sound Scenes and Events

Guillaume Lemaitre, Nicolas Grimault, and Clara Suied

Abstract Auditory scenes are made of several different sounds overlapping in time and frequency, propagating through space, and resulting in complex arrays of acoustic information reaching the listeners' ears. Despite the complexity of the signal, human listeners *segregate* effortlessly these scenes into different *meaningful sound events*. This chapter provides an overview of the auditory mechanisms subserving this ability. First, we briefly introduce the major characteristics of sound production and propagation and basic notions of psychoacoustics. The next part describes one specific family of auditory scene analysis models (how listeners segregate the scene into auditory objects), based on multidimensional representations of the signal, temporal coherence analysis to form auditory objects, and the attentional processes that make the foreground pop out from the background. Then, the chapter reviews different approaches to study the perception and identification of sound events (how listeners make sense of the auditory objects): the identification of different properties of sound events (size, material, velocity, etc.), and a more general approach that investigates the acoustic and auditory features subserving sound recognition. Overall, this review of the acoustics and psychoacoustics of sound scenes and events provides a backdrop for the development of computational methods reported in the other chapters of this volume.

Keywords Psychoacoustics • Acoustics • Auditory perception • Auditory scene analysis • Sound and auditory events • Auditory object • Sound identification • Features

G. Lemaitre (✉)

STMS-Ircam-CNRS-UPMC, 1 place Stravinsky, 75004 Paris, France

e-mail: GuillaumeJLemaitre@gmail.com

N. Grimault

UMR CNRS 5292, Centre de Recherche en Neurosciences de Lyon, Université Lyon 1,

50 Av. Tony Garnier, 69366 Lyon Cedex 07, France

e-mail: nicolas.grimault@cnrs.fr

C. Suied

Institut de Recherche Biomédicale des Armées, Brétigny-sur-Orge, France

e-mail: clarasuied@gmail.com

3.1 Introduction

Everyday situations are rarely silent. In most situations, sound-producing events are constantly happening, overlapping in time and frequency, propagating through space, reflecting from surfaces, and being diffracted by obstacles, before finally merging at a microphone or the listener's ear. This complex acoustic array constitutes the *auditory scene*. Imagine, for example, yourself relaxing at home with a few friends and kids. In this case, the auditory scene is extremely cluttered and busy, composed of different conversations and noises. Yet, listeners are able to parse the scene, segregate different *sound events* from a background, identify these sound events (different conversations, background music, the clinking of glasses, a suspicious bang from the kid's playground), follow the conversation they are taking part in, switch their attention to a potentially more interesting conversation that is happening nearby, monitor the kids' antics, etc. This is the classical *cocktail party effect* [4, 97, 124]. The goal of this chapter is to sketch the physics underlying these sounds, and to provide an overview of a few basic psychoacoustic and cognitive models subserving the ability to process, parse, and make sense of the sound events composing an auditory scene.

Whereas the perception of speech and music has been studied for a long time, the study of the perception of everyday scenes and events (non-speech, non-music sounds occurring in a daily environment) is relatively new. The main characteristic of everyday sound perception ("everyday listening") is that its primary goal is to make sense of what is happening in a listener's environment, by segregating the scene into different events and identifying the events. Music perception ("musical listening"), in comparison, is more focused on the musical qualities of the signals, and not so much on the precise identification of the sound sources [30]. Speech perception finally is about decoding a linguistic message and identifying the identity, gender, emotion of speakers.

The acoustic signals reaching the listeners' ears in an auditory scene are first processed by the peripheral auditory system (pinna, eardrum, ossicles, cochlea, auditory nerve). The perception of auditory scenes and events involves in addition two important perceptual abilities: segregation and grouping of spectro-temporal regularities in the acoustic signal into auditory objects (*auditory scene analysis*); Making sense of the auditory objects, i.e., identifying the sound events they correspond to (*sound event identification*). An auditory object is a percept (a mental entity resulting from the perception of a phenomenon) corresponding to a sound (or a group of sounds) perceived as a coherent whole, e.g., the complex sequence of noises emitted by a printer that is assigned (correctly or incorrectly) to a single source or event in the environment (e.g., a printer printing out pages) [4, 7, 11].

This chapter reports on the current status of research addressing these three aspects. Section 3.2 first briefly introduces the acoustics of auditory scenes, as well as the models of peripheral auditory processing used as a front-end of many models of auditory scene analysis and sound event identification. Then Sect. 3.3 uses the framework of one model family of auditory scene analysis to overview several

important perceptual phenomena involved in scene analysis (multidimensional representation, temporal coherence). Section 3.4 reviews the literature on sound event identification. As such, this subsection will review some studies dedicated to evidence the effects of knowledge, attention, or multisensory integration to auditory scene analysis.

3.2 Acoustic and Psychoacoustic Characteristics of Auditory Scenes and Events

This section first introduces important features of a sound signal (amplitude, periodicity, frequency, spectrum, etc.), how sounds are produced and propagate, and basic notions of how sounds are perceived (psychoacoustics). To do so, it first presents a simplified model of peripheral auditory processing (outer/middle ear, cochlea, neural transduction) and the basic auditory percepts (pitch, loudness, dimensions of timbre) that mirror the acoustic characteristics previously mentioned.

3.2.1 Acoustic Characteristics of Sound Scenes and Events

A sound is produced by some mechanical vibration in contact with the air (e.g., the soundboard of a guitar) or a rapid modulation of air flow (e.g., the vocal folds periodically interrupting the expulsion of air from the lungs when speaking or singing). These phenomena create oscillations of the pressure of the air that propagate through this medium. These *sound waves* can be converted back into physical motion by the eardrum or the diaphragm of a microphone, and are generally represented as a *waveform*, shown in panel (c) of Fig. 3.1, where the y -axis represents the *amplitude* of the pressure (generally represented in arbitrary units) and the x -axis represents time. The level (L) of a sound is often represented in decibels (dB): $L_{\text{dB}} = 20 \log_{10}(\text{RMS})$, where RMS is the root mean square value of the signal. An important class of sounds, including speech and music consists of waveforms repeating periodically over a certain amount of time (see panel (d) of Fig. 3.1). The duration of an elementary waveform is called the *period* (in s), and the inverse of the period is the *frequency* (in Hz). Other sounds do not repeat periodically, but are made of random variations of air pressure. These are called noises (see below). By application of the Fourier transform, any waveform can be decomposed into a sum of elementary sinusoids at different frequencies. A common representation of a sound signal is to plot its Fourier transform, as represented in panel (a) of Fig. 3.1 (with frequency represented on the y -axis). The representation of the Fourier transform is called the *spectrum* of the signal. The Fourier transform can also be computed over short overlapping time windows (the short-time Fourier transform), thus providing a time-frequency representation of sound signals: its magnitude is represented as the *spectrogram* (represented in panel (b) of Fig. 3.1).

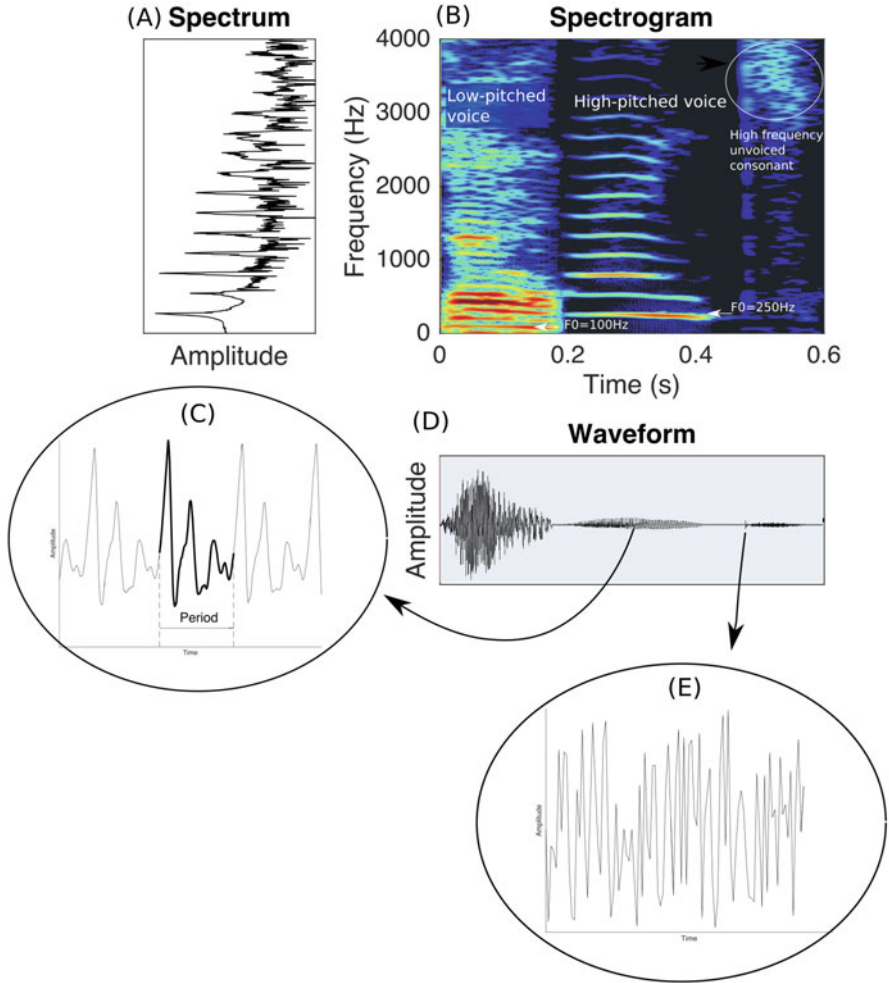


Fig. 3.1 Example of a voice signal (one word pronounced by a male speaker followed by a word pronounced by a female speaker). Panel (b) represents the spectrogram of the signal (x-axis is time, y-axis is frequency, color codes amplitude). Panel (a) represents the spectrum of the female speaker. Panel (c) represents the waveform. A closer look [panel (d)] at the waveform of the female speaker shows that it consists of the repetition of an elementary waveform. A close-up on a noisy consonant [panel (e)] shows that the waveform consists of random variations

3.2.1.1 Periodic and Non-periodic Signals

The class of sounds that can be described as multiple repetitions of an elementary waveform (at least approximately) are called *periodic* signals. For example, Fig. 3.1 represents a speech signal, consisting of a first word uttered by a male voice followed by a second word uttered by a female voice. Vowels are often periodic

in the middle of their duration. The Fourier transform of a periodic signal is made of a sum of pure tones (the “partials”) whose frequencies form a *harmonic series*. The frequency of the lowest tone is called the *fundamental frequency* (F_0). It is approximately 100 Hz for the male voice and 250 Hz for the female voice shown in Fig. 3.1. The frequencies of the other partials are integer multiples of the fundamental (F_0 , $2F_0$, $3F_0$, etc., see the spectrogram of the female voice at the top of Fig. 3.1). They are called the *harmonics* of the signal. The periodicity and the fundamental frequency of a sound are related to the sensation of pitch, through a complex relationship (see Sect. 3.2.2.2, “Pitch and loudness”). Most periodic sounds occurring in a daily environment are produced by animal vocalizations (including human speech), musical instruments, or human-made artifacts with rotating parts (e.g., car engines).

In contrast, many non-human everyday sounds are not periodic. *Non-periodic sounds* do not create any sensation of pitch. For example, rubbing together two pieces of wood and waves breaking on a shore produce sounds that are non-periodic. Sounds resulting from a random process (such as aerodynamic turbulences created by wind gusts) are another example of non-periodic sounds called *noisy*. In the speech domain, consonants are usually noisy (“unvoiced”—see, for example, the consonant represented in panel (b) of Fig. 3.1; it clearly lacks the harmonic structure of the vowels). Panel (e) of Fig. 3.1 represents a close-up on the waveform of a consonant, showing the random fluctuations of air pressure.

3.2.1.2 Sound Production and Propagation

Sounds are usually produced by one or several physical objects set in vibration by an action executed on that object. For example, banging on a door causes the door to vibrate, and this vibration is transmitted (“radiated”) to surrounding air. Many physical systems have linear regimes that support privileged vibration frequencies, determined by their geometry. These are the *modes of vibration*. For example, a string fixed at both ends (the most simple vibrating system) vibrates at frequencies that form a harmonic series, and F_0 is proportional to the inverse of the length of the string. The sounds produced by an object set in vibration result from a combination of these modes. The amplitude of each mode depends on how the object is set in vibration, where it is struck, etc. For example, a string fixed at both ends and plucked at a point one third of the way along its length will miss the $3F_0$, $6F_0$ modes, etc. Other kinds of excitation may produce noisy sounds. For example, rubbing together two pieces of wood creates a very dense and random series of micro impacts, which results in a sound that lack any harmonic structure (i.e., noises).

Sounds propagate through air as pressure waves. The speed of sound in air is approximately 343 m/s at 20°C. Close to the sound source, spherical radiation leads to a pressure wave whose amplitude in linear units is inversely proportional to distance. The attenuation in dB is proportional to the log of the distance (drops by 6 dB for every doubling of distance). Farther away from the source, sound waves can be considered as plane waves, and the attenuation (in dB) is proportional to the

distance to the source and to the square of the frequencies: as sounds travel through space, they become quieter and higher frequencies are attenuated. In free field (i.e., when sound waves travel freely without being reflected or diffracted by obstacles), the sound pressure level and the frequency content of a sound could be used to estimate the distance of a sound source. In most situations, however, the original spectrum of sounds is unknown, sound sources move, and sound waves rarely travel in free field.

In fact, sound waves also encounter obstacles as they travel. Depending on the size of the obstacles, sound waves are diffracted (they “bend over”) or reflected by the obstacles. An important case occurs when waves propagate in enclosed spaces and bounce off walls. In that case, the sound wave created by the sound source will reach a microphone or the listener’s ears through multiple paths: the wave will first reach the microphone via the shortest path, but also later via multiple reflections from the walls, floor, and ceiling of the space. In large spaces, early reflections will be clearly separated from the direct sound (similar to an echo), and later reflections will be merged together and interfere. The sum of these reflections is called the “reverberation” (or room effect). The sound that reaches a listener’s ear or a microphone is thus different from the sound that would be picked up close to the source. These multiple paths can be modeled by the impulse response from the source to the microphone. In addition, the comparison of the sound waves reaching the microphone through direct and indirect paths can be used to estimate the distance to a sound source [57].

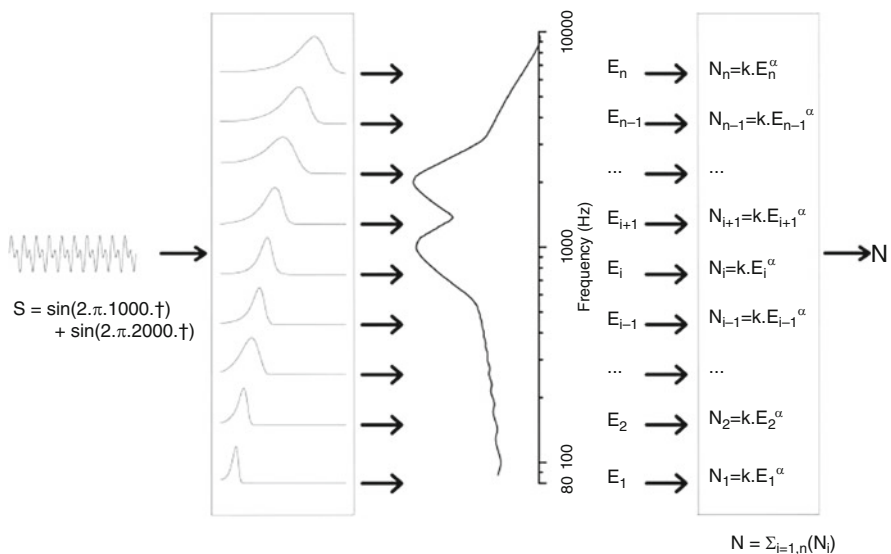
When multiple sources are present, the sound waves generated by these sources (and their reflections) will merge by simply summing the air pressure variations. However, such summing can result in interference: Imagine, for instance, adding up two pure tones at the same frequency. If the two pure tones are shifted in time by half a period (phase opposition), the positive part of one sine wave will add to the negative part of the other wave, thus resulting in no sound (destructive interference). Sound do not simply add as the sum of their magnitude spectra: phase delays will create interferences that can profoundly modify the spectrum of the sound sources (and thus their timbre). The challenge for the auditory system is to disentangle the mixture that reaches the listener’s ears to recover the individual sound sources.

3.2.2 Psychoacoustics of Auditory Scenes and Events

Psychoacoustics aims at establishing quantitative relationships between percepts and acoustical properties of the signals. Pitch and loudness are probably the best-known percepts. Others percepts are collectively denominated as “timbre.” This section first introduces basic models of how the peripheral auditory system processes sounds. Then, we describe loudness and pitch and some models (“acoustic correlates”) of these percepts. Finally, we review the notion of timbre. These models form the basis of the features used in many auditory scene analysis systems. A more detailed account of the features used in computational systems is provided in Chap. 4.

3.2.2.1 Models of Peripheral Auditory Processing

Models of the outer (pinna and ear canal), middle (eardrum, auditory ossicles, and oval window), and inner ears (cochlea) have existed for a long time. The outer and middle ears can be simply simulated by a bandpass filter with a gain curve following the minimum auditory field or other equal-loudness contours. The inner ear is modeled by a bank of auditory filters that approximate the basilar membrane motion followed by a model of neural firing to convert the motion of the basilar membrane into a pattern of neural activity. More precisely, an auditory filter is a linear bandpass filter designed to approximate data from psychoacoustical and neurophysiological studies. An excitation pattern (see Fig. 3.2) is a representation of the energy at the output of a bank of auditory filters with central frequencies nonlinearly distributed in the audible frequency range. The shape and the bandwidth of the auditory filters are still a matter of debate. Several mathematical models of the auditory filters have been proposed and implemented in a wealth of applications, ranging from the simplest (the Bark scale [127]) to more complicated models defined in the spectral (e.g., the roexp filter family) or temporal domains (e.g., the gammatone and gammachirp filter families [115]). Similarly, many mathematical



Input \rightarrow Outer, middle and inner ear filtering \rightarrow Excitation pattern \rightarrow Partial loudness \rightarrow Loudness

Fig. 3.2 Example of the framework used by several models to compute global loudness (N) from a two-tone signal S . The first step corresponds to the filtering by the outer, middle, and inner ears with a bank of auditory filters, resulting in an excitation pattern E_i defined in each frequency band i . Partial loudness N_i can then be computed from the excitation pattern using a compressive law and summed to estimate the global loudness

models of neural transduction have been proposed, from the simplest (half-wave rectification, compression, and low-pass filtering) to physiological models based on inner-hair cell and synapse properties [106, 126]. For example, the latest models of inner-hair cells and auditory nerves can now account for the spontaneous activity in the auditory-nerve fibers [106].

3.2.2.2 Pitch and Loudness

Pitch and loudness are the percepts that have been the most studied. Pitch is the percept that enables listeners to order sounds from low to high on a musical scale. For pure tones, pitch can be simply related to frequency. More generally, pitch is related to periodicity cues for other complex sounds, and many authors have been interested in modeling pitch perception [23].

For pure tones, the perception of pitch is generally assumed to be based both on tonotopical cues (i.e., the place of excitation along the basilar membrane in the cochlea) and phase locking cues (i.e., the temporal correlation between the phase of the input signal and the discharge rate of the neurons) up to a spectral limit from a few kHz to 5 kHz or even higher. Over this spectral limit, the inner-hair cells cannot discharge as fast as necessary neither to follow the phase of the signal nor to even be correlated with it. Temporal cues become then ineffective and only spectral cues are available [45].

To explain the perception of pitch evoked by harmonic complex tones, autocorrelative models that extract periodicities of the excitation in the auditory nerve [67] are into the limelight as they can account for many experimental results from the literature. However, such unitary models of pitch perception cannot account for all data in the literature [39] and dual models taking both the periodicity and the place of excitation should enable a better understanding of pitch perception as suggested by Oxenham et al. [78]. In practical applications, YIN [24], Praat [12], and Straight [48] are probably the three most used pitch detection algorithms (for monophonic pitch).

Loudness is the percept that enables listeners to order sounds on a scale from quiet to loud. For pure tones, loudness is simply related to sound pressure by a power law, whose exponent depends on the frequency of the tone [101]. For complex sounds, however, loudness depends on the intensity and the spectrum of the sound, via complex nonlinear relationships. Several models have been proposed and normalized over the years and are now commonly used in commercial applications [70, 73, 128]. Most models share a common framework showed in Fig. 3.2: loudness can be estimated by integrating partial loudness along the frequency bands and partial loudness can be estimated either by adding a compressive stage to the basilar membrane motion at the output of each auditory filter or directly from the auditory nerve response pattern. Recent developments have refined the shape of the auditory filters, the summation procedures, and modified the models to account for the loudness of very short, very quiet, or time-varying sounds (e.g., [15, 86]).

3.2.2.3 The Dimensional Approach to Timbre

One commonly used definition of timbre is summarized by Risset and Wessel: “the perceptual attribute that enables us to distinguish among orchestral instruments¹ that are playing the same pitch, and are equally loud” [5, 91]. In this approach, timbre is not considered as a single percept, as loudness or pitch. It is multidimensional: it consists of several percepts (or dimensions) and the goal of the psychoacoustical approach to timbre has precisely been to characterize the dimensions of timbre.

A widespread method to study the dimensions of timbre uses dissimilarity ratings and multidimensional scaling (MDS) techniques, in a three-step procedure. First, listeners rate the dissimilarities between each pair of sounds of a set. Second, MDS techniques represent the dissimilarity data by distances in a geometrical space (*perceptual space*), wherein the distance between two points represents the dissimilarity between two sounds. It is assumed that the dimensions of the space represent independent percepts. These dimensions are then interpreted by acoustical parameters (called in this case *acoustical correlates*). This approach is based on the assumption of common continuous dimensions shared by the sounds.

There is a long tradition of studies of the timbre of musical instruments [17, 38, 64]. More recently, the technique has been adapted and applied to a variety of sounds of industrial products [108]: air conditioning units, car horns, car doors, wind buffeting noise [55, 80, 107]. Several dimensions have been found quite systematically across this wide range studies (see, for example, [69] for a meta-analysis): sharpness, roughness and fluctuation strength, attack time, tonalness, spectral flux, odd-to-even ratio. The following paragraphs detail three of these dimensions and their acoustic correlates. The acoustical correlates reported here are computed on the basis of models of peripheral auditory processing reported in Sect. 3.2.2. In short, they correspond to some statistical properties of the auditory signal, either purely in the temporal or spectral domains [127]. They are practically available in many commercial or freely distributed toolboxes (e.g., the “MIR Toolbox” [53], the “Timbre Toolbox” [84], Head Acoustics’ Artemis,² B&K’s Pulse³). Note, however, that some more recent models also exist that use joint spectro-temporal representations [27].

Sharpness and Brightness

Sharpness and brightness correspond to a similar percept: that sounds can be ordered on a scale ranging from dull to sharp or bright. It is correlated with the spectral balance of energy: sounds with a lot of energy in low frequencies are perceived as dull whereas sounds with a lot of energy in high frequencies are perceived as bright or sharp. The acoustical correlate of brightness is the spectral centroid. It is calculated as the barycenter (i.e., first statistical moment) of the Fourier transform

¹Note that this definition does only apply to musical instruments, though.

²<https://www.head-acoustics.de>.

³<https://www.bksv.com/>.

of the acoustic signal. There are many variations to this formula. For example, Zwicker's sharpness is calculated as the barycenter of the output of the auditory filter bank, weighted by a function that emphasizes high frequencies (Zwicker's sharpness is thus in fact a measurement of the amount of high frequencies) [127].

Fluctuation Strength and Roughness

Fluctuation strength and roughness both correspond to the perception of modulations of the amplitude of the signal, but different ranges of modulation frequency result in two different percepts. When modulations are slow (around 4 Hz), the sounds are perceived as fluctuating (wobbling): This is the percept of fluctuation strength. Faster modulations (around 70 Hz) are perceived as rough (harsh): This is the percept of roughness.

Zwicker and Fastl have proposed an acoustic correlate of fluctuation strength [127], calculated by estimating the modulation depth and modulation frequency in each Bark band. Note that this method does not work very well for noisy signals. In that case, the method described by [100] is better suited.

The calculation of an acoustic correlate to the percept of roughness cannot be simply described. A commonly used algorithm is described in [20]. It consists of calculating an index of modulation in 47 channels (overlapping Bark band), and then summing the contribution of each channel, weighted by the cross-correlation of the signal's envelope in adjacent channels.

Onset

Onset is the percept related to the time a sound takes to start. Onset corresponds to a sensory continuum ranging from slow onsets (e.g., bowed strings, sanding a piece of wood) to impulsive sounds (e.g., plucked strings, knocking on a door). Onset is best correlated with the logarithm of the attack time [84]. There are different methods to estimate the attack time. The most common method consists of using a fixed threshold (e.g., 10% of the maximum of the envelope, see Fig. 3.3). However, Peeters et al. have developed a more robust method ("weakest effort") [83].

3.3 The Perception of Auditory Scenes

To attend to sound events in natural environments, the auditory system is generally confronted with the challenge of analyzing the auditory scene and parsing the scene into several auditory objects. Ideally, each auditory object corresponds to a distinct sound source. To decide to group or to segregate sounds, one theory of perceptual organization suggests that the auditory system applies rules named the "Gestalt laws of grouping": *Proximity*, *similarity*, *good continuation*, and *common fate*. *Proximity* groups together sounds coming from the same spatial position. *Similarity* groups together sounds sharing some perceptual features (e.g., pitch, timbre). *Good continuation* considers that a new sound is occurring when an abrupt change occurs and *common fate* groups together sounds that are congruent in time. This theory is still a matter of debate and remains an open question.

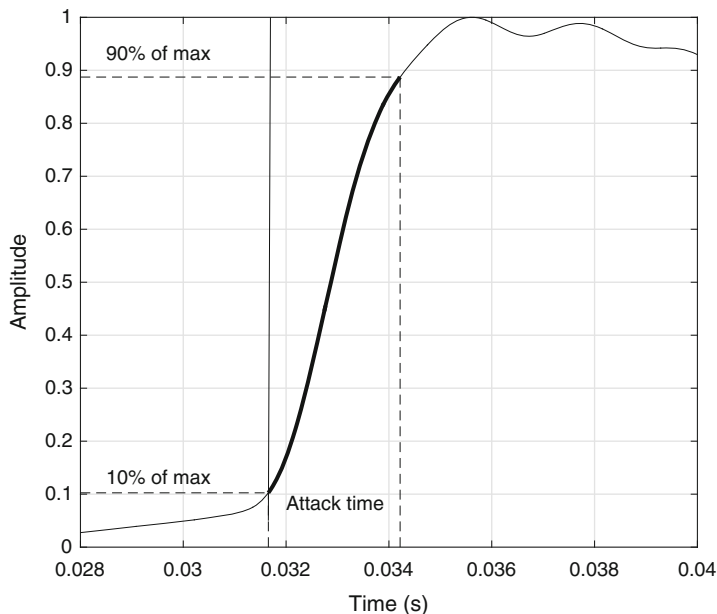


Fig. 3.3 The calculation of the attack time. In this example, the attack time is calculated between the times where the envelope of the signal increases from 10 to 90% of its maximal value [83]

Computational auditory scene analysis (CASA) is a wide field of research that has been successively and extensively reviewed by Cooke and Ellis [19], Wang and Brown [121], and more recently Szabó et al. [109]. In this field, several approaches coexist based on different theoretical basis. Szabó et al. [109] have classified the models of auditory scene analysis in three categories labeled as Bayesian, neural, and temporal coherence models. Bayesian models use Bayesian inferences to update continuously the probability of occurrence of auditory events. Neural models are based on neural networks with excitatory and inhibitory neurons. Each auditory event corresponds then to a population of neurons that interact with the others. Finally, temporal coherence models are based on the idea that a sound event can be characterized by stabilized acoustic features with some coherent fluctuations. Bayesian and neural models are discussed in Chaps. 2 and 5. Here, we focus on the particular model family, based on a temporal coherence approach.

These specific models of auditory scene analysis [26, 50, 95] are based on three stages. First a multidimensional representation of the signal is computed as a function of time with each axis corresponding to an auditory dimension. Second, this multidimensional space is interpreted in terms of auditory objects using a temporal coherence analysis. The general idea is to group all parts of the signal that are temporally coherent along one or more dimensions. The last stage involves attentional processes to make the foreground pop out from the background.

3.3.1 *Multidimensional Representation*

Any salient perceptual difference [71] and its corresponding acoustic correlates can be potentially used to segregate sounds from each other's [72]. Pitch, timbre dimensions, or temporal dynamics (see Sect. 3.2.2) are prominent but the contribution of any other percepts (related, for example, to loudness differences [116] or spatial position of the sound source) can also contribute to this perceptual organization [21, 68]. As these acoustic correlates can be independently expressed as functions of time, they can then be seen as axes of a multidimensional perceptual space, with time as one of the dimensions. The Gestalt laws of grouping then apply to this perceptual space. *Proximity* and *similarity* laws are simply related to a perceptual distance in this space.

3.3.2 *Temporal Coherence*

In this perceptual space, the time dimension is rather special and is characteristic of sound perception: Sounds unfold in time and require time to be perceived, which is not the case for visual stimuli. As such, the Gestalt laws of *good continuation* and *common fate* can be interpreted in terms of temporal coherence across or within the other perceptual dimensions. Temporal coherence within a perceptual dimension is related to the average correlation of this perceptual dimension between two different instances in time, where the strength of the effect decreases with increasing time separation. This section will review how temporal coherence influences grouping and how this has been modeled. The first attempt to assess the effect of temporal coherence with sequences of pure tones was to vary the time delay between tones of different frequencies f_1, f_2 [116]. The general idea was that a change from f_1 to f_2 in the signal will be less incoherent if the change is smooth ($f_2 - f_1$ is small) or if the time delay that occurs between the change increases. Many studies reviewed in Bregman's book [13] evidence that when the time delay increases (increasing the cross-frequency coherence), the degree of segregation decreases. These studies only involve pure tones of two different frequencies but this result generalizes to more realistic sounds as vowels differing in fundamental frequencies [29]. Early computational models defined temporal coherence as an onset/offset coincidence detector across frequency bands (for example, in Brown and Cook's model [14]) whereas more recent models propose an integrated model of temporal coherence across and within all dimensions [95]. In this case, coherence is continuously computed as the cross-correlation coefficients between the frequency channel responses integrated over a time windows between 50 and 500 ms. For example, in panel (b) of Fig. 3.1, the channels containing the harmonics of 100 Hz will be highly coherent during the first milliseconds (low pitched voice) and grouped together as an integrated auditory object while the channels containing the harmonics of 250 Hz will be highly coherent during the high pitched voice. At the transition point

between the voices, the bands centered on multiples of 100 Hz and those centered on multiples of 250 Hz will not be coherent and segregation will occur. Such a coherence analysis can apply simultaneously to each perceptual dimension, and each dimension can be allocated with a particular weight to guide the segregation decision. In some extreme situations, incoherent signals can even be integrated in the same auditory object. For example, the high-frequency unvoiced consonant in panel (b) of Fig. 3.1, which is pronounced by the same speaker, should be perceptively linked to the preceding high pitched voice, as recently evidenced [22]. The strong incoherences in the frequency channels should then be ignored (allocated with a weight close to zero) and coherences in other perceptual dimensions should then dominate the segregation/grouping process. For now, rules to tell these situations apart are not known and remain to be defined by further studies.

3.3.3 Other Effects in Segregation

Since the publication of Bregman's book [13], the concept of *schema-based segregation* has been largely acknowledged in the literature. A schema can be seen as a memory trace related to a well-known sound event and this intuitive concept of schema involves numerous cognitive processes that could all contribute to segregation. As such, this subsection will review some studies dedicated to evidence the effects of knowledge, attention, or multisensory integration to auditory scene analysis. It is possible to view the concept of schema as an aspect of the temporal coherence that has been previously described. In fact, in the temporal coherence analysis, the importance of the temporal coherence of a single dimension (i.e., the allocated weight) could be enhanced by predictability based for example, on knowledge, attention, or other higher-level processes. As illustrated by Shamma, segregating a female voice from a background could imply to give more importance to the sounds coming from a particular position (because you can see where the female speaker is) or to high-frequency regions of the spectrum because you can predict that the voice will be high-pitched [95]. The effect of predictability from the experimental point of view has been reviewed by Bendixen et al. [8]. For example, Bendixen et al. [9, 10] showed that the predictability of a frequency pattern of pure tones can promote segregation, whereas other authors showed an effect of temporal regularities (temporal predictability) on segregation [25, 90]. From the modeling point of view, some authors also integrate a prediction-driven approach. For example, in the context of speech/non-speech mixtures, Ellis [28] has proposed a model architecture that compares and reconciles the stimulus-driven extracted speech signal with the words predicted by a knowledge-based word-model. Another overlooked aspect of auditory scene analysis is the contribution of congruent visual cues. From the early 1950s, it was well established that lip reading improves speech recognition in noise [105]. However, it is unclear if the improvement is due to the amount of phonetic information contained in the visual cue per se or if the congruent visual cues enhanced the segregation process. Few studies addressed this question

[94]. When the visual cues contain some information about another feature useful for segregation (i.e., pitch variation), a clear effect of visual cues on segregation has been reported [62]. This effect of the visual cues on segregation remains significant but the effect size is, however, strongly reduced when the visual cues, providing only congruent temporal information, do not provide any helpful cues by themselves for segregation [25].

3.4 The Perception of Sound Events

As well as parsing an auditory scene into distinct sound sources, listeners also have to perform the closely related task of identifying the individual sources. In fact, humans are remarkably able to recognize complex and natural sound sources reliably and without any apparent effort. We report on two approaches that have investigated this ability. A first approach (“psychomechanics”) has focused on different properties of the physical events causing sounds (e.g., material, size, velocity) and sought the acoustic correlates subserving the identification of these properties. Another more global approach has sought to identify minimal features in biologically inspired representations of auditory signals that allow sound recognition.

3.4.1 *Perception of the Properties of Sound Events: Psychomechanics*

Psychomechanics studies the perception of the physical properties of sound sources [63]. In fact, mechanical sounds are produced by an action executed on an object or several objects interacting among themselves. In principle, listeners can therefore perceive both the properties of the objects (e.g., size, shape) and actions (e.g., type of action, speed, force). For example, Chap. 7 describes how cognitive representations of these properties are organized, with certain properties being more accessible than some others. Here, this section describes the results of studies that have investigated how accurately listeners identify these different properties, and the acoustic correlates that subserve their identification.

3.4.1.1 Material

Material has probably been the most-studied causal property of sounds [51]. Some researchers have searched for an acoustic feature unique to each material which does not vary despite changes in other object and action properties such as shape, size, and force. Wildes and Richards were the first to propose such a feature for

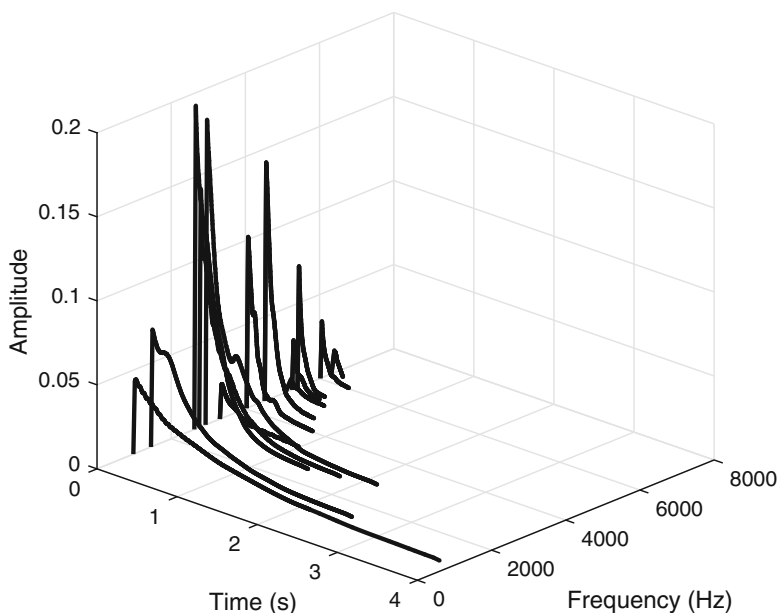


Fig. 3.4 Calculation of the coefficient of internal friction. This figure represents the partials of a metal pipe being impacted [54]. The decay of each partial decreases with frequency (lower frequencies ring for a longer time than higher frequencies). The rate of decay with frequency is characteristic of each material [123]

anelastic linear solids: the coefficient of internal friction $\tan \phi$ [123] (see Fig. 3.4). This coefficient determines the damping of vibration in a solid object. They showed that this coefficient could be estimated by first measuring the time t_e required by the amplitude of each partial to decrease to $1/e$ of its starting value: $\tan \phi = 1/(\pi f t_e)$ (where f is the frequency of the partial). Then defining $\alpha(f)$ as the decay of the sound over time (the inverse of t_e , in s^{-1}), the decay of the partials with frequency is modeled with a linear function whose parameter is the coefficient of internal friction: $\alpha(f) = \tan \phi \pi f$.

The coefficient of internal friction is characteristic of each material. However, several studies have shown that human listeners' sensitivity to this feature is limited [59]. Listeners are generally only able to distinguish coarse categories of material from sounds: they can tell metal from wood, but not metal from glass or wood from plastic [31, 113]. In fact, listeners do not use only the damping: they also use purely spectral features such as the frequency of the first partial, which are inaccurate predictors of the material [49, 65].

3.4.1.2 Shape and Size

The question of whether it is possible to hear the shape of an object first interested mathematicians [33, 46]. In contrast to material perception, the conclusions of the studies of the auditory perception of the size and shape of objects are less clear. Regarding *shape*, Lakatos et al. have shown that listeners can discriminate the sounds of impacted bars of different sections, and that their judgments were based on the frequencies from torsional vibrational and transverse bending modes⁴ [52]. Similar results were found for impacted plates of different shapes [51]. Lutfi et al. showed that the combination of decay rate, frequency, and amplitude of partials could be used in theory to determine whether an object is hollow or solid, but that listeners are not necessarily adopting optimal strategies, thus resulting in inaccurate identification [58].

Several studies have also sought to find the acoustic correlates of the *size* of an object [35, 36, 42, 76]. The major issue is that a sound is often produced by two objects in interaction. For example, the sound of ball hitting a plate is produced by both the ball and the plate resonating. Furthermore, other parameters of the interaction also influence acoustic parameters. For example, the sound of a ball dropped on a plate is louder when the ball is larger, but also when the ball is dropped from a greater height. Overall these studies have shown that listeners use two main types of cues to determine the size of the objects: loudness and spectral content (e.g., spectral centroid). The weighting of these cues depends on the particular context, and varies from individual to individual. For example, Giordano et al. have shown that the perceptual weight of acoustic features depends on whether they are informative, but also on the ability of the listeners to exploit the features [32].

3.4.1.3 Parameters of Actions

The last result is not surprising when considering that listeners are actually much more accurate at identifying the actions than the objects that have caused the sounds [54]. Surprisingly, less data is available about the acoustic correlates of the actions.

Studies have shown that listeners can distinguish objects bouncing or breaking apart, based on the temporal patterns of the individual events [122]. They can also estimate the time when a receptacle filled with water will overflow, based on rate of frequency change [16], and hear the subtle characteristics of a person's gait (stride length and cadence) [125]. But the auditory perception of the **velocity** of an object in motion is probably the action parameter that has received the most interest [40], even though empirical results suggest that the auditory system is less

⁴Physical objects have some privileged vibration frequencies, determined by their geometry. These are the *modes of vibration*. The sounds produced by an object set in vibration result from a combination of these modes. The particular combination depends on how the object is set in vibration (e.g., where it is struck).

accurate than the visual system to estimate velocity [18]. There are actually two theories as to how the auditory system could estimate the velocity of an object in motion [75]. One possibility is that the auditory estimates the location of the source at different instants (“snapshots”) and compares the distance traveled and the time taken to travel that distance. Another possibility is that the auditory uses acoustic cues that directly specify azimuth and thus (indirectly) velocity: interaural temporal differences, Doppler shift,⁵ and loudness changes. Psychoacoustic studies have confirmed that listeners may use these different cues [47, 60, 93].

Overall, these results suggest that listeners strive to make sense of what they hear, and use whatever piece of information may help them. When unambiguous information is available (e.g., the coefficient of internal friction in a impact, Doppler shift, etc.), they use it. When relevant information is not fully available, they may rely on strategies that are sub-optimal in general, but that will nevertheless provide them with some coarse approximation.

3.4.2 Minimal and Sparse Features for Sound Recognition

Whereas the psychomechanics approach focuses on isolated properties of the sound sources, another line of research has sought to investigate more globally the important and necessary features of auditory representation subserving sound recognition. In short, the rationale behind this approach is that studying the ability of listeners to identify sounds sources when the sounds are degraded will highlight the features that are necessary for recognition.

The processing of natural stimuli may recruit specific mechanisms derived from adaptation to natural environments [56, 74, 111]. For example, Smith and Lewicki have developed a theoretical approach in accordance with neurophysiological data, which shows that the auditory code and especially the auditory filters (see Sect. 3.2.2) are optimal for natural sounds [99]. They have shown, among others, that all information in a sound is not necessary for its recognition.

3.4.2.1 Spectral Regions, Minimal Durations, and Spectro-Temporal Modulations

This is in fact a well-known phenomenon for speech stimuli: the auditory stimuli can be drastically distorted and modified (for example, with a noise-band vocoder method) but still remain recognizable [96]. More recently, Gygi et al. have applied the same technique to environmental sounds, and identified the important spectro-temporal regions for recognition [41]. Although less spectacular than for speech,

⁵The Doppler effect causes a dramatic change of the perceived pitch of a moving object as it passes the observer.

this study shows that, for environmental sounds as well, recognition is possible even with highly filtered sounds (same frequency regions than for speech, with slightly more high frequencies useful for environmental sounds recognition). However, the results were highly variable depending on the environmental sounds considered: for a given experimental condition, results spanned the entire range of recognition accuracy, which is easily explained by the large acoustical variability of the environmental sounds.

To avoid this drawback of the difficult study of environmental sounds, a useful distinction between pulse-resonance (like speech or melodic music) and noisy sounds has been developed by Patterson [82]. Future studies will probably benefit from this distinction, as they will probably lead to different set of characteristic features.

To deal with the question of the features useful for sound recognition in a more systematic way, the different studies have applied multiple constraints to the signal or the task in order to see how listeners can still perform the task in degraded conditions.

An obvious candidate that can be applied to an acoustic waveform is a temporal constraint (i.e., gating). This approach has mainly been used with pulse-resonance sounds (speech and music). For example, in 1942, Gray proposed the technique of “phonemic microtomy” [37]: he extracted short segments of vowels sounds, pronounced at different fundamental frequencies, and presented them to listeners. He showed that, for some vowels, recognition was possible for segments as short as 3 ms, which was less than one cycle of sound. This result was confirmed with various vowel types [88, 102]. This gating technique was also applied to the sounds of musical instruments [92]. Interestingly, the results showed that recognition was possible for durations shorter than what was required to identify the pitch or the octave of a sound. More recently, Suied et al. used the gating paradigm with a much more diverse set of musical sounds (singing voices and instruments), and a larger acoustical variability (the short segment of sound heard by the listeners was, on each trial, extracted randomly from the original sound) [104]. They confirmed that a very short segment of sound is sufficient for recognition to be above chance, with better recognition for voices (4 ms) than for musical instruments (8 ms). They have also shown, comparing their results to the prediction of the multiple-looks model [119], that, for gate durations up to 16 ms, perceptual results outperformed an ideal observer model, thus suggesting that timbre cues are available at a variety of time-scales, even very short ones [110].

Although the large majority of studies have focused on pulse-resonance sounds like speech or music, because of their direct relevance for humans, natural sounds are also often noisy: wind, rain, fire, etc. These categories of sounds have recently been studied as acoustic textures [66, 77]. Noisy natural sounds can be statistically modeled with neuroscience-inspired algorithms, on different time-scale modulations, to take into account the variety of time-scales present in the auditory analysis [114]. On a similar vein, McDermott et al. have shown that sound textures perception can be modeled with a relatively small number of summary statistics based on early auditory representations [66].

3.4.2.2 Sparse Features

All these methods can reduce drastically the signal, but they are not particularly sparse. This is in contrast with the growing body of evidence from physiology, suggesting that cortical coding of sounds may in fact be sparse [43]. Two studies have addressed this question by proposing a new method to reveal sparse features for sound recognition, using “auditory sketches” [44, 103]. Auditory sketches are sparse representations of sounds, severely impoverished compared to the original, which nevertheless afford good performance on a given perceptual task. Starting from biologically grounded representations (auditory models), a sketch is obtained by reconstructing a highly under-sampled selection of elementary “atoms” (i.e., non-zero coefficients in a given representation; here, an auditory representation). Then, the sketch is evaluated with a psychophysical experiment involving human listeners. These studies have shown that even very simplified sounds can still be recognized, although there is variability across categories of sounds. One limitation of these sketch studies is the choice of the features.

The selection of these features, or atoms, which are the dictionary elements, is obviously of prime importance. Two selection methods have been compared [103], with a slightly better result for a simple peak-picking algorithm, which kept only the larger amplitude values, compared to an analysis-based iterative thresholding method [87]. This study was a first proof of concept, but did not take a full advantage of the current major trend of signal processing for audio coding to use sparse representations [85].

An outstanding issue for these signal processing techniques is to choose the appropriate feature dictionary for sparse coding. Most of the state-of-the-art dictionary-based methods (e.g., using non-negative matrix factorization or probabilistic latent component analysis) operate on some spectral representation [120]. Other approaches (for example, the k-SVD technique [2]) attempt to learn dictionaries of elementary waveforms by maximizing some cost criteria that balances the coding cost/sparsity and the average approximation error, on a set of training signals. However, only a few current optimization criteria for building these dictionaries have been proposed that explicitly take into account human perception. This should also be explored in a near future, with the goal of being used for perception-related applications, such as hearing aids. For example, Patil et al. have reported very accurate classification of musical instrument sounds by using joint spectro-temporal representations modeling auditory cortical responses [81].

Another option would be to infer the feature set used for recognition from the behavioral data. This idea has been developed very recently for speech recognition [61, 117, 118] and musical instruments recognition [112], with techniques inspired by reverse correlation, wherein features are initially chosen randomly and then selected based on behavioral performance (the “bubble” technique used in vision [34]). These reverse correlation techniques, originally developed for feature detection in audition [3], have proven very effective to reveal important spectro-temporal modulations for speech intelligibility, and should be a good path to follow for natural sound and environmental sounds recognition in general.

Taken together, the results in this section suggest that our auditory system can retain few diagnostic features of a given sound, for recognition and further use, although it is obvious that it depends on the task, as we can still easily differentiate a sketch from its natural version, in the same way as we can differentiate a visual sketch from a photography. Perhaps through experience, listeners seem to be able to learn these discriminant features, and can then afford fast and robust recognition of sometimes very impoverished signals.

3.4.3 Discussion: On the Dimensionality of Auditory Representations

The concepts of auditory dimension and features have been used extensively throughout this chapter: timbre studies have highlighted a few systematic dimensions across a wide range of sounds, the segregation of auditory scene analysis is based on the temporal coherence of auditory dimensions, and recent approaches have shown that recognition of sound sources relies on a few features of sparse representations of the signals. It is important to note that these results do not support the idea that auditory perception could be based on a limited, reduced, and rigid set of features used in every situation. On the contrary, auditory features seemed to be task- and listener-dependent. As an example, recent studies have shown that prior exposure to complex, abstract noise textures with little structure improves the performance at a subsequent repetition detection task [1]. This suggests that auditory representations for noise encode some complex statistical properties of sounds, and that these properties may in fact be noise- and listener-dependent. Another important consideration comes from the music information retrieval community. In fact, the techniques that best recognize musical instruments or styles are not based on a few recurrent dimensions: they use and create whichever pieces statistical information are useful for a given classification task, relying on high-dimensional representations [6, 79, 81, 98]. Although this does not prove that the auditory system necessarily uses such a strategy, it shows that high-dimensional, redundant representations can yield performances similar to human listening.

Taken as a whole, these results show that, whereas similarity judgments (such as those used in timbre studies reported in Sect. 3.2.2.3) may be based on a small number of common dimensions, identifying sounds may in fact rely on diagnostic features, idiosyncratic to each sound class, learned by listeners through their individual experience [89]. In other words, these results suggest a versatile auditory system that uses whichever pieces of auditory information fit its purpose, sampled from an optimized representation of the acoustic environment. In fact, the auditory system may have evolved to provide optimized representations of the variability of the acoustic environment (including speech signals) [99]. Such representations may capture the complexity of the acoustic environment in a very effective way. A current challenge of auditory research is to discover these representations. Another challenge is to understand how the auditory system uses such representations to parse complex auditory scenes.

3.5 Summary

Auditory scenes result from a myriad of sound waves merging at a microphone or the listener's ears, originating from many sound sources, propagating through a medium (air) whose attenuation is frequency-dependent, bouncing off rigid surfaces and bending over smaller obstacles, adding one with another through complex patterns of interferences. The complex mixture at the listener's ears is thus very different from the sounds of each individual source, yet listeners make sense of these complex auditory scenes effortlessly. This ability relies on three aspects of audition: The auditory system processes and encodes incoming auditory signals, segregates the scenes into auditory objects, and associates the auditory objects with sound sources. This chapter has reviewed studies of these mechanisms as well as their current models. The peripheral auditory system is usually modeled by a bank of overlapping auditory filters followed by a nonlinear component. The outputs of such models are used to compute a variety of features correlated with basic auditory dimensions: loudness, pitch, and the dimensions of timbre. The coherence of these dimensions over time for different auditory objects then serves the purpose of segregating the complex mixture reaching a listener's ears into auditory objects. Segregation mechanisms benefit in addition from attention, prior knowledge and expectations, and multisensory integration. Listeners also associate auditory objects with their sources with no apparent effort: they identify the sound events that have caused the sounds. Research has identified the acoustic correlates of certain perceived properties of sound events (e.g., material, size, velocity), yet the perception of these properties is not accurate in every case. Recent approaches have tackled the issue of sound identification from a different perspective, seeking to identify features of biologically inspired sparse representations of auditory signals that subserve sound recognition.

References

1. Agus, T.R., Thorpe, S.J., Pressnitzer, D.: Rapid formation of robust auditory memories: insights from noise. *Neuron* **66**, 610–618 (2010)
2. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **54**(11), 4311–4322 (2006)
3. Ahumada, A. Jr., Lovell, J.: Stimulus features in signal detection. *J. Acoust. Soc. Am.* **49**, 1751–1756 (1970)
4. Alain, C., Arnott, S.R.: Selectively attending to auditory objects. *Front. Biosci.* **5**, D202–D212 (2000)
5. American Standard Association: USA acoustical terminology S1.1–160. American Standard Association (1960)
6. Aucouturier, J.J., Bigand, E.: Mel Cepstrum & Ann Ova: the difficult dialog between MIR and music cognition. In: *ISMIR*, pp. 397–402. Citeseer (2012)
7. Backer, K.C., Alain, C.: Attention to memory: orienting attention to sound object representations. *Psychol. Res.* **78**(3), 439–452 (2014)

8. Bendixen, A.: Predictability effects in auditory scene analysis: a review. *Front. Neurosci.* **8**, 60 (2014)
9. Bendixen, A., Denham, S.L., Gyimesi, K., Winkler, I.: Regular patterns stabilize auditory streams. *J. Acoust. Soc. Am.* **128**, 3658–3666 (2010)
10. Bendixen, A., Bóhm, T.M., Szalárdy, O., Mill, R., Denham, L.S., Winkler, I.: Different roles of similarity and predictability in auditory stream segregation. *Learn. Percept.* **5**(2), 37–54 (2013)
11. Bizley, J.K., Cohen, Y.E.: The what, where and how of auditory-object perception. *Nat. Rev. Neurosci.* **14**(10), 693–707 (2013)
12. Boersma, P., Weenink, D.: Praat: doing phonetics by computer (version 5.1.05) (2009). Computer program. <http://www.praat.org/>. Retrieved May 1, 2009
13. Bregman, A.S.: *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, Cambridge (1990)
14. Brown, G.J., Cooke, M.: Computational auditory scene analysis. *Comput. Speech Lang.* **8**(4), 297–336 (1994)
15. Buus, S., Müsch, H., Florentine, M.: On loudness at threshold. *J. Acoust. Soc. Am.* **104**(1), 399–410 (1998)
16. Cabe, P.A., Pittenger, J.B.: Human sensitivity to acoustic information from vessel filling. *J. Exp. Psychol. Hum. Percept. Perform.* **26**(1), 313–324 (2000)
17. Caclin, A., McAdams, S., Smith, B.K., Winsberg, S.: Acoustic correlates of timbre space dimensions: a confirmatory study using synthetic tones. *J. Acoust. Soc. Am.* **118**(1), 471–482 (2005)
18. Carlile, S., Best, V.: Discrimination of sound source velocity in human listeners. *J. Acoust. Soc. Am.* **111**(2), 1026–1035 (2002)
19. Cooke, M., Ellis, D.P.: The auditory organization of speech and other sources in listeners and computational models. *Speech Commun.* **35**(3), 141–177 (2001)
20. Daniel, P., Weber, R.: Psychoacoustical roughness: implementation of an optimized model. *Acust. United Acta Acust.* **83**, 113–123 (1997)
21. David, M., Lavandier, M., Grimault, N.: Sequential streaming, binaural cues and lateralization. *J. Acoust. Soc. Am.* **138**(6), 3500–3512 (2015)
22. David, M., Lavandier, M., Grimault, N., Oxenham, A.: Sequential stream segregation of voiced and unvoiced speech sounds based on fundamental frequency. *Hear. Res.* **344**, 235–243 (2017)
23. de Cheveigné, A.: Pitch perception models. In: Plack, C., Oxenham, A. (eds.) *Pitch*, chap. 6, pp. 169–233. Springer, New York (2004)
24. de Cheveigné, A., Kawahara, H.: YIN, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.* **111**(4), 1917–1930 (2002)
25. Devergie, A., Grimault, N., Tillmann, B., Berthommier, F.: Effect of rhythmic attention on the segregation of interleaved melodies. *J. Acoust. Soc. Am.* **128**, EL1–EL7 (2010)
26. Elhilali, M., Ma, L., Micheyl, C., Oxenham, A.J., Shamma, S.: Temporal coherence in the perceptual organization and cortical representation of auditory scenes. *Neuron* **61**(2), 317–329 (2009)
27. Elliott, T.M., Hamilton, L.S., Theunissen, F.E.: Acoustic structure of the five perceptual dimensions of timbre in orchestral instrument tones. *J. Acoust. Soc. Am.* **133**(1), 389–404 (2013)
28. Ellis, D.P.: Using knowledge to organize sound: the prediction-driven approach to computational auditory scene analysis and its application to speech/nonspeech mixtures. *Speech Commun.* **27**(3), 281–298 (1999)
29. Gaudrain, E., Grimault, N., Healy, E., Béra, J.: Effect of spectral smearing on the perceptual segregation of vowel sequences. *Hear. Res.* **231**, 32–41 (2007)
30. Gaver, W.W.: What do we hear in the world? An ecological approach to auditory event perception. *Ecol. Psychol.* **5**(1), 1–29 (1993)
31. Giordano, B.L., McAdams, S.: Material identification of real impact sounds: effect of size variation in steel, glass, wood and plexiglass plates. *J. Acoust. Soc. Am.* **119**(2), 1171–1881 (2006)

32. Giordano, B.L., McAdams, S., Rocchesso, D.: Integration of acoustical information in the perception of impacted sound sources: the role of information accuracy and exploitability. *J. Exp. Psychol. Hum. Percept. Perform.* **36**(2), 462–476 (2010). doi:10.1037/a0018388
33. Gordon, C., Webb, D.: You can't hear the shape of a drum. *Am. Sci.* **84**(1), 46–55 (1996)
34. Gosselin, F., Schyns, P.G.: Bubbles: a technique to reveal the use of information in recognition tasks. *Vis. Res.* **41**(17), 2261–2271 (2001)
35. Grassi, M.: Do we hear size or sound? Balls dropped on plates. *Percept. Psychophys.* **67**(2), 274–284 (2005)
36. Grassi, M., Pastore, M., Lemaitre, G.: Looking at the world with your ears: how do we get the size of an object from its sound? *Acta Psychol.* **143**, 96–104 (2013)
37. Gray, G.W.: Phonemic microtomy: the minimum duration of perceptible speech sounds. *Commun. Monogr.* **9**(1), 75–90 (1942)
38. Grey, J.M., Moorer, J.A.: Perceptual evaluation of synthesized musical instrument tones. *J. Acoust. Soc. Am.* **62**, 454–462 (1977)
39. Grimault, N., Micheyl, C., Carlyon, R., Collet, L.: Evidence for two pitch encoding mechanisms using a selective auditory training paradigm. *Percept. Psychophys.* **64**(2), 189–197 (2002)
40. Guski, R.: Acoustic Tau: an easy analogue to visual Tau? *Ecol. Psychol.* **4**(3), 189–197 (1992)
41. Gygi, B., Kidd, G.R., Watson, C.S.: Spectral-temporal factors in the identification of environmental sounds. *J. Acoust. Soc. Am.* **115**(3), 1252–1265 (2004)
42. Houben, M.M., Kohlrausch, A., Hermes, D.J.: The contribution of spectral and temporal information to the auditory perception of the size and speed of rolling balls. *Acta Acust. United Acust.* **91**, 1007–1015 (2005)
43. Hromádka, T., Zador, A.M.: Representations in auditory cortex. *Curr. Opin. Neurobiol.* **19**(4), 430–433 (2009)
44. Isnard, V., Taffou, M., Viaud-Delmon, I., Suied, C.: Auditory sketches: very sparse representations of signals are still recognizable. *PLoS One* **11**(3), e0150313 (2016)
45. Joris, P.X., Verschooten, E.: On the limit of neural phase locking to fine structure in humans. *Basic Asp. Hear.* **787**, 101–108 (2013)
46. Kac, M.: Can one hear the shape of a drum? *Am. Math. Mon.* **73**(4), 1–23 (1966)
47. Kaczmarek, T.: Auditory perception of sound source velocity. *J. Acoust. Soc. Am.* **117**(5), 3149–3156 (2005)
48. Kawahara, H., Masuda-Katsuse, I., De Cheveigne, A.: Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f0 extraction: possible role of a repetitive structure in sounds. *Speech Commun.* **27**(3), 187–207 (1999)
49. Klatzky, R.L., Pai, D.K., Krotkov, E.P.: Perception of material from contact sounds. *Presence* **9**(4), 399–410 (2000)
50. Krishnan, L., Elhilali, M., Shamma, S.: Segregating complex sound sources through temporal coherence. *PLoS Comput. Biol.* **10**, e1003985 (2014)
51. Kunkler-Peck, A.J., Turvey, M.T.: Hearing shape. *J. Exp. Psychol. Hum. Percept. Perform.* **26**(1), 279–294 (2000)
52. Lakatos, S., McAdams, S., Caussé, R.: The representation of auditory source characteristics: simple geometric forms. *Percept. Psychophys.* **59**(8), 1180–1190 (1997)
53. Lartillot, O., Toivianen, P., Eerola, T.: A Matlab toolbox for music information retrieval. In: Preisach, C., Burkhardt, H., Schmidt-Thieme, L., Decker, R. (eds.) *Data Analysis, Machine Learning and Applications*, pp. 261–268. Springer, Berlin (2008)
54. Lemaitre, G., Heller, L.M.: Auditory perception of material is fragile, while action is strikingly robust. *J. Acoust. Soc. Am.* **131**(2), 1337–1348 (2012)
55. Lemaitre, G., Susini, P., Winsberg, S., Letinturier, B., McAdams, S.: The sound quality of car horns: a psychoacoustical study of timbre. *Acoust. United Acta Acust.* **93**(3), 457–468 (2007)
56. Lewicki, M.S.: Efficient coding of natural sounds. *Nat. Neurosci.* **5**(4), 356–363 (2002)

57. Lu, Y.C., Cooke, M.: Binaural estimation of sound source distance via the direct-to-reverberant energy ratio for static and moving sources. *IEEE Trans. Audio Speech Lang. Process.* **18**(7), 1793–1805 (2010)
58. Lutfi, R.A.: Auditory detection of hollowness. *J. Acoust. Soc. Am.* **110**(2), 1010–1019 (2001)
59. Lutfi, R.A., Stoelinga, C.N.J.: Sensory constraints on auditory identification of the material and geometric properties of struck bars. *J. Acoust. Soc. Am.* **127**(1), 350–360 (2010)
60. Lutfi, R.A., Wang, W.: Correlational analysis of acoustic cues for the discrimination of auditory motion. *J. Acoust. Soc. Am.* **106**(2), 919–928 (1999)
61. Mandel, M.I., Yoho, S.E., Healy, E.W.: Measuring time-frequency importance functions of speech with bubble noise A. *J. Acoust. Soc. Am.* **140**(4), 2542–2553 (2016)
62. Marozeau, J., Innes-Brown, H., Grayden, D., Burkitt, A., Blamey, P.: The effect of visual cues on auditory stream segregation in musicians and non-musicians. *PLoS One* **5**(6), e11297 (2010)
63. McAdams, S.: The psychomechanics of real and simulated sound sources. *J. Acoust. Soc. Am.* **107**(5), 2792–2792 (2000)
64. McAdams, S., Winsberg, S., Donnadieu, S., Soete, G.D., Krimphoff, J.: Perceptual scaling of synthesized musical timbres: common dimensions, specificities and latent subject classes. *Psychol. Res.* **58**, 177–192 (1995)
65. McAdams, S., Chaigne, A., Roussarie, V.: The psychomechanics of simulated sound sources: material properties of impacted bars. *J. Acoust. Soc. Am.* **115**(3), 1306–1320 (2004)
66. McDermott, J.H., Simoncelli, E.P.: Sound texture perception via statistics of the auditory periphery: evidence from sound synthesis. *Neuron* **71**(5), 926–940 (2011)
67. Meddis, R., O’Mard, L.: A unitary model of pitch perception. *J. Acoust. Soc. Am.* **102**(3), 1811–1820 (1997)
68. Middlebrooks, J.C., Onsan, Z.A.: Stream segregation with high spatial acuity. *J. Acoust. Soc. Am.* **132**(6), 3896–3911 (2012)
69. Misdariis, N., Minard, A., Susini, P., Lemaitre, G., McAdams, S., Parizet, E.: Environmental sound perception: meta-description and modeling based on independent primary studies. *EURASIP J. Speech Audio Music Process.* **2010** (2010). Article ID 362013
70. Moore, B.C.: Development and current status of the “Cambridge” loudness models. *Trends Hear.* **18**, 2331216514550620 (2014)
71. Moore, B., Gockel, H.: Factors influencing sequential stream segregation. *Acoust. United Acta Acoust.* **88**, 320–333 (2002)
72. Moore, B.C.J., Gockel, H.E.: Properties of auditory stream formation. *Philos. Trans. R. Soc. B* **367**, 919–931 (2012)
73. Moore, B.C.J., Glasberg, B.R., Baer, T.: A model for the prediction of thresholds, loudness and partial loudness. *J. Audio Eng. Soc.* **45**(4), 224–238 (1997)
74. Nelken, I., Rotman, Y., Yosef, O.B.: Responses of auditory-cortex neurons to structural features of natural sounds. *Nature* **397**(6715), 154–157 (1999)
75. Neuhoff, J.G.: Auditory motion and localization. In: Neuhoff, J.G. (ed.) *Ecological Psychoacoustics*, pp. 87–111. Brill, Leiden (2004)
76. O’Meara, N., Bleeck, S.: Size discrimination of transient sounds: perception and modelling. *J. Hearing Sci.* **3**(3), 32–44 (2013)
77. Overath, T., Kumar, S., Stewart, L., von Kriegstein, K., Cusack, R., Rees, A., Griffiths, T.D.: Cortical mechanisms for the segregation and representation of acoustic textures. *J. Neurosci.* **30**(6), 2070–2076 (2010)
78. Oxenham, A.J., Bernstein, J.G.W., Penagos, H.: Correct tonotopic representation is necessary for complex pitch perception. *Proc. Natl. Acad. Sci.* **101**(5), 1421–1425 (2004)
79. Pachet, F., Roy, P.: Analytical features: a knowledge-based approach to audio feature generation. *EURASIP J. Audio Speech Music Process.* **2009**(1), 1 (2009)
80. Parizet, E., Guyader, E., Nosulenko, V.: Analysis of car door closing sound quality. *Appl. Acoust.* **69**, 12–22 (2008)
81. Patil, K., Pressnitzer, D., Shamma, S., Elhilali, M.: Music in our ears: the biological bases of musical timbre perception. *PLoS Comput. Biol.* **8**(11), e1002759 (2012)

82. Patterson, R.D.: Pulse-resonance sounds. In: *Encyclopedia of Computational Neuroscience*, pp. 2541–2548. Springer, New York (2015)
83. Peeters, G.: A large set of audio features for sound description (similarity and classification) in the CUIDADO project. *Cuidado projet report*, Institut de Recherche et de Coordination Acoustique Musicque (IRCAM), Paris (2004)
84. Peeters, G., Giordano, B.L., Susini, P., Misdariis, N., McAdams, S.: The timbre toolbox: extracting audio descriptors from musical signals. *J. Acoust. Soc. Am.* **130**(5), 2902 (2011)
85. Plumbley, M.D., Blumensath, T., Daudet, L., Gribonval, R., Davies, M.E.: Sparse representations in audio and music: from coding to source separation. *Proc. IEEE* **98**(6), 995–1005 (2010)
86. Ponsot, E., Susini, P., Meunier, S.: A robust asymmetry in loudness between rising-and falling-intensity tones. *Atten. Percept. Psychophys.* **77**(3), 907–920 (2015)
87. Portilla, J.: Image restoration through l0 analysis-based sparse optimization in tight frames. In: *2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 3909–3912. IEEE, New York (2009)
88. Powell, R.L., Tosi, O.: Vowel recognition threshold as a function of temporal segmentations. *J. Speech Lang. Hear. Res.* **13**(4), 715–724 (1970)
89. Pressnitzer, D., Agus, T., Suied, C.: Acoustic timbre recognition. In: *Encyclopedia of Computational Neuroscience*, pp. 128–133. Springer, Berlin (2015)
90. Rajendran, V.G., Harper, N.S., Willmore, B.D., Hartmann, W.M., Schnupp, J.W.H.: Temporal predictability as a grouping cue in the perception of auditory streams. *J. Acoust. Soc. Am.* **134**, EL98–EL104 (2013)
91. Risset, J.C., Wessel, D.L.: Exploration of timbre by analysis and synthesis. In: Deutsch, D. (ed.) *The Psychology of Music*, Series in Cognition and Perception, 2nd edn. pp. 113–169. Academic, New York (1999)
92. Robinson, K., Patterson, R.D.: The duration required to identify the instrument, the octave, or the pitch chroma of a musical note. *Music Percept. Interdiscip. J.* **13**(1), 1–15 (1995)
93. Rosenblum, L.D., Carello, C., Pastore, R.E.: Relative effectiveness of three stimulus variables for locating a moving sound source. *Perception* **16**(2), 175–186 (1987)
94. Schwartz, J.L., Grimault, N., Hupé, J.M., Moore, B.C.J., Pressnitzer, D.: Introduction: multistability in perception: binding sensory modalities, an overview. *Philos. Trans. R. Soc. B* **367**, 896–905 (2012)
95. Shamma, S.A., Elhilali, M., Micheyl, C.: Temporal coherence and attention in auditory scene analysis. *Trends Neurosci.* **34**(3), 114–123 (2011)
96. Shannon, R.V., Zeng, F.G., Kamath, V., Wygonski, J., Ekelid, M.: Speech recognition with primarily temporal cues. *Science* **270**(5234), 303 (1995)
97. Shinn-Cunningham, B.G.: Object-based auditory and visual attention. *Trends Cogn. Sci.* **12**(5), 182–186 (2008)
98. Siedenburg, K., Fujinaga, I., McAdams, S.: A comparison of approaches to timbre descriptors in music information retrieval and music psychology. *J. New Music Res.* **45**(1), 27–41 (2016)
99. Smith, E.C., Lewicki, M.S.: Efficient auditory coding. *Nature* **439**(23), 978–982 (2006)
100. Sontacchi, A.: *Entwicklung eines modulkonzeptes für die psychoakustische geräuschenanalyse unter matlab* (1999). Diplomarbeit, Institut für Elektronische Musik der Kunstuniversität Graz, Graz
101. Stevens, S.S., Galanter, E.H.: Ratio scales and category scales for a dozen of perceptual continua. *J. Exp. Psychol.* **54**(6), 377–411 (1957)
102. Suen, C.Y., Beddoes, M.P.: Discrimination of vowel sounds of very short duration. *Percept. Psychophys.* **11**(6), 417–419 (1972)
103. Suied, C., Drémeau, A., Pressnitzer, D., Daudet, L.: Auditory sketches: sparse representations of sounds based on perceptual models. In: Aramaki, M., Barthet, M., Kronland-Martinet, R., Ivi Ystad, S. (eds.) *From Sounds to Music and Emotions*, 9th International Symposium, CMMR 2012, London, June 19–22, 2012, Revised Selected Papers. *Lecture Notes in Computer Science*, vol. 7900, pp. 154–170. Springer, Berlin (2013)

104. Suied, C., Agus, T.R., Thorpe, S.J., Mesgarani, N., Pressnitzer, D.: Auditory gist: recognition of very short sounds from timbre cues. *J. Acoust. Soc. Am.* **135**(3), 1380–1391 (2014)
105. Sumbly, W.H., Pollack, I.: Visual contribution to speech intelligibility in noise. *J. Acoust. Soc. Am.* **26**, 212–215 (1954)
106. Sumner, C., Lopez-Poveda, E., O'Mard, L., Meddis, R.: A revised model of the inner-hair cell and auditory-nerve complex. *J. Acoust. Soc. Am.* **111**(5), 2178–2188 (2002)
107. Susini, P., McAdams, S., Winsberg, S., Perry, I., Vieillard, S., Rodet, X.: Characterizing the sound quality of air-conditioning noise. *Appl. Acoust.* **65**(8), 763–790 (2004)
108. Susini, P., Lemaitre, G., McAdams, S.: Psychological measurement for sound description and evaluation. In: Berglund, B., Rossi, G.B., Townsend, J.T., Pendrill, L.R. (eds.) *Measurement with Persons - Theory, Methods and Implementation Area*, chap. 11. Psychology Press/Taylor and Francis, New York (2011)
109. Szabó, B.T., Denham, S.L., Winkler, I.: Computational models of auditory scene analysis: a review. *Front. Neurosci.* **10**, 524 (2016)
110. Teng, X., Tian, X., Poeppel, D.: Testing multi-scale processing in the auditory system. *Sci. Rep.* **6** (2016). doi:10.1038/srep34390
111. Theunissen, F.E., Sen, K., Doupe, A.J.: Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds. *J. Neurosci.* **20**(6), 2315–2331 (2000)
112. Thoret, E., Depalle, P., McAdams, S.: Perceptually salient spectrotemporal modulations for recognition of sustained musical instruments. *J. Acoust. Soc. Am.* **140**(6), EL478–EL483 (2016)
113. Tucker, S., Brown, G.J.: Modelling the auditory perception of size, shape and material: applications to the classification of transient sonar sounds. In: *Audio Engineering Society Convention*, vol. 114 (2003). <http://www.aes.org/e-lib/browse.cfm?elib=12543>
114. Turner, R., Sahani, M.: Modeling natural sounds with modulation cascade processes. In: *Advances in Neural Information Processing Systems*, pp. 1545–1552 (2008)
115. Unoki, M., Irino, T., Glasberg, B., Moore, B., Patterson, R.: Comparison of the roex and gammachirp filters as representations of the auditory filter. *J. Acoust. Soc. Am.* **120**(3), 1474–1492 (2006)
116. van Noorden L.: *Temporal coherence in the perception of tone sequences*. Ph.D. thesis, Eindhoven University of Technology (1975)
117. Varnet, L., Knoblauch, K., Meunier, F., Hoen, M.: Using auditory classification images for the identification of fine acoustic cues used in speech perception. *Front. Hum. Neurosci.* **7**, 865 (2013)
118. Venezia, J.H., Hickok, G., Richards, V.M.: Auditory “bubbles”: efficient classification of the spectrotemporal modulations essential for speech intelligibility. *J. Acoust. Soc. Am.* **140**(2), 1072–1088 (2016)
119. Viemeister, N.F., Wakefield, G.H.: Temporal integration and multiple looks. *J. Acoust. Soc. Am.* **90**(2), 858–865 (1991)
120. Virtanen, T., Gemmeke, J.F., Raj, B., Smaragdis, P.: Compositional models for audio processing: uncovering the structure of sound mixtures. *IEEE Signal Process. Mag.* **32**(2), 125–144 (2015)
121. Wang, D., Brown, G.J.: *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, New York (2006)
122. Warren, W.H., Verbrugge, R.R.: Auditory perception of breaking and bouncing events: a case study in ecological acoustics. *J. Exp. Psychol. Hum. Percept. Perform.* **10**(5), 704–712 (1984)
123. Wildes, R.P., Richards, W.A.: Recovering material properties from sound. In: Richards, W.A. (ed.) *Natural Computation*. A Bradford Book, chap. 25, pp. 356–363. The MIT Press, Cambridge, MA (1988)
124. Winkler, I., Denham, S.L., Nelken, I.: Modeling the auditory scene: predictive regularity representations and perceptual objects. *Trends Cogn. Sci.* **13**(12), 532–540 (2009)
125. Young, W., Rodger, M., Craig, C.M.: Perceiving and reenacting spatiotemporal characteristics of walking sounds. *J. Exp. Psychol. Hum. Percept. Perform.* **39**(2), 464–476 (2012)

126. Zilany, M., Bruce, I., Nelson, P., Carney, L.: A phenomenological model of the synapse between the inner hair cell and auditory nerve: long-term adaptation with power-law dynamics. *J. Acoust. Soc. Am.* **126**(5), 2390–2412 (2009)
127. Zwicker, E., Fastl, H.: *Psychoacoustics Facts and Models*, 463 pp. Springer, Berlin (1990)
128. Zwicker, E., Fastl, H., Widmann, U., Kurakata, K., Kuwano, S., Namba, S.: Program for calculating loudness according to DIN 45631 (ISO 532B). *J. Acoust. Soc. Jpn.* **12**(1) (1991). doi10.1250/ast.12.39

Part II

Core Methods

Chapter 4

Acoustic Features for Environmental Sound Analysis

Romain Serizel, Victor Bisot, Slim Essid, and Gaël Richard

Abstract Most of the time it is nearly impossible to differentiate between particular type of sound events from a waveform only. Therefore, frequency-domain and time-frequency domain representations have been used for years providing representations of the sound signals that are more in line with the human perception. However, these representations are usually too generic and often fail to describe specific content that is present in a sound recording. A lot of work has been devoted to design features that could allow extracting such specific information leading to a wide variety of hand-crafted features. During the past years, owing to the increasing availability of medium-scale and large-scale sound datasets, an alternative approach to feature extraction has become popular, the so-called feature learning. Finally, processing the amount of data that is at hand nowadays can quickly become overwhelming. It is therefore of paramount importance to be able to reduce the size of the dataset in the feature space. The general processing chain to convert a sound signal to a feature vector that can be efficiently exploited by a classifier and the relation to features used for speech and music processing are described in this chapter.

Keywords Feature extraction • Feature engineering • Audio signal representation • Audio signal processing • Time-frequency representation • Multiscale representation • Representation learning • Perceptually motivated features • Feature selection • Dimensionality reduction • Feature pooling • Temporal integration

R. Serizel (✉)

Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, France
e-mail: romain.serizel@loria.fr

V. Bisot • S. Essid • G. Richard

LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France
e-mail: victor.bisot@telecom-paristech.fr; slim.essid@telecom-paristech.fr;
gael.richard@telecom-paristech.fr

4.1 Introduction

The time domain representation of a sound signal, or waveform, is not easy to interpret directly. Most of the time it is nearly impossible, from a waveform, to identify or even localise sound events (unless they occur at different dynamic range, e.g., a loud noise in a quiet environment) and to discriminate between sound scenes. Therefore, frequency-domain representations and time-frequency domain representations (including multiscale representations) have been used for years providing representations of the sound signals that are more in line with the human perception.

However, these representations are usually too generic and often fail to describe specific content that is present in a sound recording. A lot of work has been devoted to design features that could allow extraction of such specific information, leading to a wide variety of hand-crafted features. One problem with these types of features is that, by design, they are specific to a task and that they usually do not generalise well. They often need to be combined with other features, leading to large feature vectors. During the past years, owing to the increasing availability of medium-scale and large-scale sound datasets, an alternative approach to feature extraction has become popular, the so-called feature learning that has proven competitive with most finely tuned hand-crafted features.

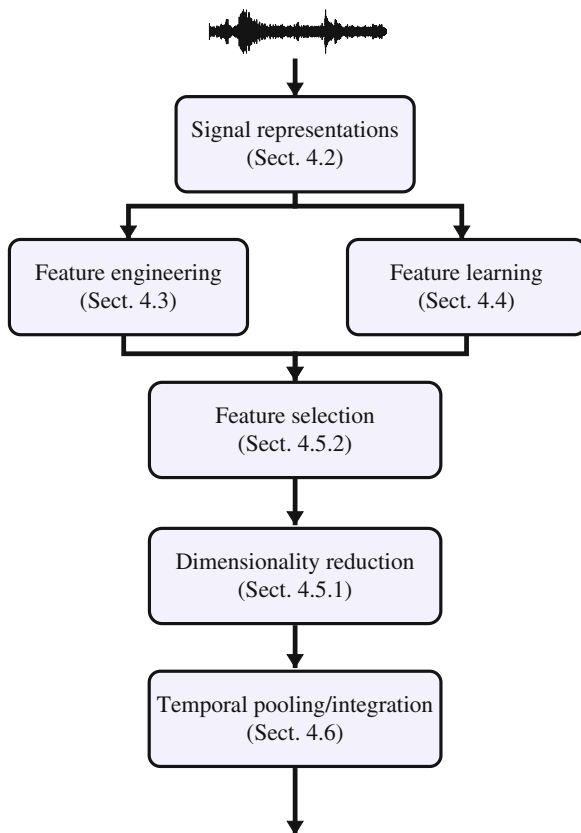
Finally, in both cases, using either feature engineering or feature learning, processing the amount of data that is at hand nowadays can quickly become overwhelming. It is therefore of paramount importance to be able to reduce the size of the dataset in the feature space either by reducing the feature vectors dimensionality or by reducing the amount of feature vectors to process.

The general processing chain to convert a sound signal to a feature vector that can be efficiently exploited by a classifier is described in this chapter. The standard steps are presented sequentially (see also Fig. 4.1). It is also crucial to design features that are robust to perturbation. Therefore, the possibility to enhance signals or enforce robustness at each step is discussed in the corresponding section when applicable. Finally, the relation to features used for speech and music processing is briefly discussed in Sect. 4.7 and conclusions are presented in Sect. 4.8.

4.2 Signal Representations

Over the years, a large amount of work has been devoted to finding appropriate representations that allow extraction of useful information from sound signals. Some of the main classes of sound signals representations are presented in this section.

Fig. 4.1 Standard feature extraction process



4.2.1 Signal Acquisition and Preprocessing

In general terms, sound is the result of a vibration that propagates as waves through a medium such as air or water. Sounds can be recorded under the form of an electric signal $x(t)$ by means of an electroacoustic transducer such as a microphone. This analog signal $x(t)$ can then be converted to a digital signal $x[n]$ and stored on a computer before further analysis. The necessary steps to perform this analog-digital conversion include:

- **A filtering stage:** the analog signal $x(t)$ is low-pass filtered in order to limit its frequency bandwidth in the interval $[0, B]$ where B is the cut-off frequency of the low-pass filter.
- **A sampling stage:** the low-passed analog signal is then digitally sampled at a sampling rate $f_s = 2B$ to avoid the well-known frequency aliasing phenomenon.
- **A quantification stage:** the obtained digital signal is then quantised (e.g. the amplitude of the signal can only take a limited number of predefined values to preserve storage capacity).

- **Optional additional stage:** in some cases, additional preprocessing stages can be performed such as pre-emphasis. This step can be performed under the form of a simple first order finite impulse response (FIR) high-pass filter. Historically, this step was performed on speech signals prior to linear prediction (LP) analysis to cope with its typical -6 dB spectral tilt which was shown to be detrimental for LP parameters estimation. In other situations, this step is less justified and is therefore not mandatory.

Typical values for audio CD quality are a sampling rate of $f_s = 44.1$ kHz and a quantisation on 16 bits per sample leading to a bit rate of 705,600 kbit/s for a single channel audio signal. Higher quality standards include sampling rates of 48, 96 or 192 kHz and quantisation on 24 bits.

4.2.2 General Time-Frequency Representations

The sound signals are usually converted to the frequency-domain prior to any analysis. The frequency-domain representation of a signal $x[n]$ on a linear-frequency scale can be obtained with the discrete-time Fourier transform (DFT):

$$X[f] = \sum_{n=-\infty}^{\infty} x[n]e^{-i2\pi fn} \quad (4.1)$$

The spectrum $X[f]$ is f_s -periodic in f with f_s the sampling frequency. The frequency $f = \frac{f_s}{2}$ represents the Nyquist-frequency.

The spectrum $X[f]$ can be transformed back to time domain with the inverse discrete-time Fourier transform (IDFT):

$$x[n] = \frac{1}{f_s} \int_{-\frac{f_s}{2}}^{\frac{f_s}{2}} X[f]e^{i2\pi fn} df \quad (4.2)$$

In practice, the spectrum $X[f]$ is approximated by applying the DFT on a windowed frame of length N of the signal $x[n]$. This is referred to as the short-time Fourier transform (STFT). The f th component of the DFT of the t th frame of $x[n]$ is computed as follows:

$$X[t, f] = \sum_{k=0}^{N-1} w[k]x[tN + k]e^{\frac{-i2\pi kf}{N}} \quad (4.3)$$

where $w[k]$ is a window function (e.g. rectangular, Hamming, Blackman, etc.) used to attenuate some of the effects of the DFT approximation and to enforce continuity and periodicity at the edge of the frames. Equation (4.3) is given with a hop between frames equal to the length of the frames (N). This means that there is no overlap

between consecutive frames. It is common to choose a hop size that is smaller than the frame length in order to introduce overlap that allows for smoother STFT representation and introduces statistical dependencies between frames.

The t th frame of time domain signal $x[n]$ can be obtained from the discrete spectrum $X[t, f]$ by applying the inverse STFT. Both the STFT and the inverse STFT can be efficiently computed using the fast Fourier transform (FFT) and the inverse fast Fourier transform (IFFT), respectively.

The STFT allows for defining the linear-frequency *spectrogram* which is a 2D representation of a sound where energy in each frequency band is given as a function of time. The spectrogram is then the matrix where each column is the modulus of the DFT of a sound signal frame (see also Fig. 4.2b).

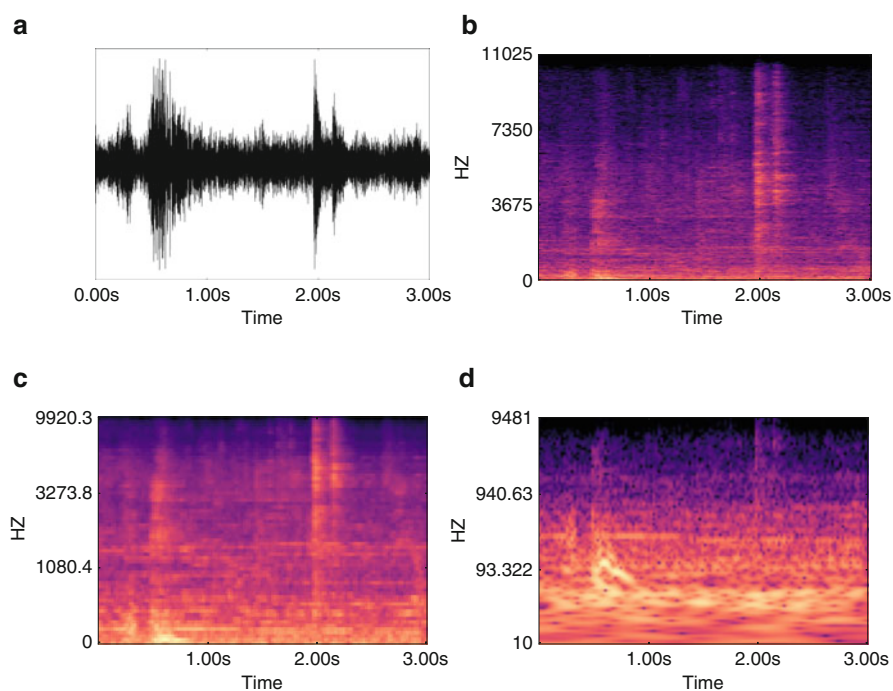


Fig. 4.2 Different time domain and time-frequency domain representations of a sound signal recorded in a restaurant: at 0.5s someone is clearing his throat, at 2s there is some cutlery noises [66]. (a) Temporal waveform. (b) Linear-frequency spectrogram. (c) Mel spectrogram. (d) Constant-Q spectrogram

4.2.3 Log-Frequency and Perceptually Motivated Representations

It is often desirable to search for information in specific frequency bands. This may be achieved by computing energy or energy ratios in predefined frequency bands (see also Fig. 4.2c). The bands can be equally spaced on the frequency axis, placed according to logarithm or perceptual laws. The number of bands, the shape of the prototype filter and the overlap between bands can also vary greatly.

1. **Critical bands** were introduced by Fletcher [33]. The key idea is that critical bands describe the bandwidth of the auditory filters in the cochlea. Conceptually, this means that two tones within the same critical band will interfere with each other, this is the so-called frequency masking phenomenon. The equivalent rectangular bandwidth scale (ERB) provides a way to compute the central frequency and bandwidth of the rectangular filters approximating the auditory filters [35]:

$$\text{ERB}(f) = 24.7 \times \left(4.37 \frac{f}{1000} + 1 \right) \quad (4.4)$$

with f in Hertz. The Bark scale is another scale relying on the concept of critical bands but that was derived from different experiments [95].

2. **Gammatone filters** are linear filters whose impulse response $\text{gamma}[n]$ is composed of a sinusoidal carrier wave (a tone) modulated in amplitude by an envelope that has the same form as a scaled gamma distribution function:

$$\text{gamma}[n] = a n^{\gamma-1} e^{-2\pi b n} \cos(2\pi f_c n + \Phi), \quad (4.5)$$

where a is the amplitude, γ is the filter order, b is a temporal decay coefficient (related to the bandwidth of the filter), f_c the frequency of the carrier (related to centre frequency of the filter) and Φ the phase of the carrier (related to the position of the envelope on the carrier). Similarly to ERB, gammatone filters of order 4 have been shown to provide a good approximation to auditory filters [71].

3. **Mel scale** corresponds to an approximation of the psychological sensation of heights of a pure sound (e.g. a pure sinusoid) [86]. Several analytical expressions exist [68], a common relation between the mel scale $\text{mel}(f)$ and the Hertz scale f was given by Fant [31]:

$$\text{mel}(f) = \frac{1000}{\log 2} \log \left(1 + \frac{f}{1000} \right) \quad (4.6)$$

4. **Constant-Q transform (CQT)** is closely related to DFT. One major difference is that instead of using a frequency scale with constant spacing between frequencies (as in DFT), the frequencies are distributed geometrically [13]. This yields a constant ratio Q between the central frequency of a band f_k and the frequency

resolution $f_k - f_{k-1}$, therefore the name CQT. The central frequency for the k th band is given by:

$$f_k = f_0 \times 2^{\frac{k}{b}}, \quad (4.7)$$

with f_0 the central frequency of the first band and b the number of frequencies per octave (see also Fig. 4.2d). This transform was originally introduced to map the western musical scale.

4.2.4 Multiscale Representations

Multiscale approaches allow for flexible decompositions, representing the sound signal on multiple scales for both time and frequency. Some of the most common approaches are presented below :

1. **Pyramids** are multiscale representations that were originally introduced for image processing [14]. Pyramids are built recursively by applying at each step a convolutive operation (filtering) followed by a down-sampling operation on a signal. This procedure allows to extract information at different resolutions. There are two main type of pyramids: the so-called Gaussian pyramids (where a low-pass filtering is applied) [14] and the Laplacian pyramids (where a band-pass filtering is applied) [15]. Pyramids with quadratic mirror filters (QMF) [22] have been shown to be closely related to wavelets [58].
2. **Wavelets** are functions that can generally be visualised as a brief oscillation and that should integrate to zero [36, 59]. Given a discrete-time wavelet $\Psi(x)$, it is possible to define a wavelet basis by applying translation and dilatation on the wavelet

$$\Psi_{ab}(x) = \frac{1}{\sqrt{a}} \Psi \left(\frac{x-b}{a} \right), \quad (4.8)$$

with $a \in \mathbb{R}^+$ the dilatation factor and $b \in \mathbb{R}$ the translation factor. The translation then allows for covering different time instants while the dilatation of the wavelet enables multiscale analysis [58]. Note that in practice a and b often take their value in a discrete subspace of \mathbb{R} , defining the so-called discrete wavelets bases.

3. **Scattering transform** builds invariant and stable representations by cascading a wavelet transform, a modulus operation and a low-pass filtering operation [60]. Scattering transform can capture non-stationary behaviour and can be interpreted as an operation that calculates modulation spectrum coefficients of multiple orders. This approach can enable the modelling of signal dynamics as well as sound textures that are important aspects in the characterisation of environmental sounds.

4.2.5 Discussion

Time-frequency representations such as STFT were designed mainly according to mathematical rules leading, for example, to linear-frequency scales. Human perception studies have shown that we do not perceive sound similarly in each region of the spectrum and that the resolution of the human ear also varies along the frequency axis. Therefore, non-linear-frequency scales have been introduced in an attempt to mimic human perception and provide a better way to extract information from sound signals. The frequency scale can be tuned to map the auditory filters (critical bands, ERB, bark scale), to match perceptual behaviour (mel scale) or according to the intrinsic properties of the signal to represent (CQT). In any case, adjusting the granularity of the frequency scale usually allows designing more accurate representations of the signal of interest and can therefore lead to increased robustness. It is also possible to apply standard frequency-domain filtering [29, 39, 93] to time-frequency domain representations in order to attenuate the effects of additive perturbations.

Perceptually motivated time-frequency representation often constitutes an important part of sound scene and event analysis systems. They serve, either as a way to visually observe the time-frequency content of the sound scene, or as an input representation to more complex classification systems. Therefore, in many cases, their computation is one of the first steps for applying some of the feature engineering or feature learning techniques presented in Sects. 4.3 and 4.4. Extracting representations based on mel or gammatone filterbanks can be necessary to compute cepstral features (see Sect. 4.3.3), which are widely popular in the field [73, 90]. Other representations such as the CQT are often used to build time-frequency images from which image-based features are extracted [10, 78, 94]. Such representations are also considered as inputs to feature learning techniques such as nonnegative matrix factorisation [6, 11, 21], or can be directly used as features for deep neural network-based systems [70, 75].

Yet, in these approaches there is only one fixed frequency scale that is non-linear and the time scale remains linear. As sound signals contain information at different time and frequency scales, parts of the signal might be overlooked with these representations. Some works based on variants of the scattering transform proved the usefulness of multiscale representations to perform sound event classification in real-life conditions [56, 82].

4.3 Feature Engineering

Similarly to other sound processing tasks, feature extraction for sound scene and event analysis has often relied on the so-called feature engineering. This is the art of carefully crafting ad-hoc features from low-level representations heavily relying on expert knowledge about class invariances. Some of the most common feature classes are presented in this section (see also Fig. 4.3).

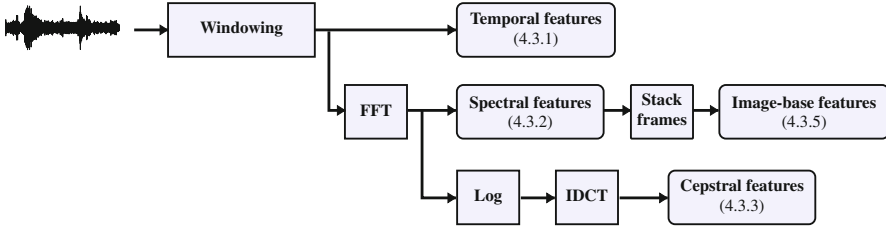


Fig. 4.3 Feature engineering process

4.3.1 Temporal Features

These features are computed directly on the temporal waveform and are therefore usually rather straightforward to compute. Some of the most common temporal features are described below.

1. **Time domain envelope** can be seen as the boundary within which the signal is contained. A simple implementation relies on the computation of the root mean square of the mean energy of the signal $x[n]$ within a frame t of size N spanning over the time indexes $n \in \{n_t, n_t + 1, \dots, n_t + N\}$:

$$e(t) = \sqrt{\frac{1}{N} \sum_{n=n_t}^{n_t+N} x[n]^2}.$$

It is a reliable indicator for silence detection.

2. **Zero crossing rate (ZCR)** is given by the number of time the signal amplitude crosses the zero value. For a frame t of size N , it is given by:

$$z_{cr}(t) = \frac{1}{2} \sum_{n=n_t}^{n_t+N} |\text{sign}(x[n]) - \text{sign}(x[n-1])|, \quad (4.9)$$

where $\text{sign}(x[n])$ returns the sign of the signal amplitude $x[n]$.

It is a very popular feature since it can, in a simple manner, discriminate periodic signals (small ZCR values) from signals corrupted by noises that are random to a certain degree (high ZCR values).

3. **Temporal waveform moments** allow the representation of different characteristics of the shape of the time domain waveform. They are defined from the first four central moments and include the following characteristics:

- centre of gravity of the waveform: *temporal centroid*,
- spread around the mean value: *temporal width*,
- waveform asymmetry around its mean: *temporal asymmetry*,
- overall flatness of the time domain waveform: *temporal flatness*.

Note that these moments can also be computed on the spectrum (see below).

4. **Autocorrelation coefficients** can be interpreted as the signal spectral distribution in the time domain. In practice, it is common to only consider the first K coefficients which can be obtained as:

$$R(k) = \frac{\sum_{n=0}^{N-k-1} x[n]x[n+k]}{\sqrt{\sum_{n=0}^{N-k-1} x^2[n]} \sqrt{\sum_{n=0}^{N-k-1} x^2[n+k]}}$$

4.3.2 Spectral Shape Features

Studies on the perception of sound widely rely on the frequency content of sound signals. Therefore, it is a natural choice to derive features from frequency representations of a signal, for example, its spectrogram. Some of the most common spectral features are described below.

1. **Energy** is one of the most straightforward yet important spectral feature. This feature can be computed directly as a sum of the squared amplitude components $|X[t, f]|$ in the band. It is also common to compute the log energy in a band.
2. **Spectral envelope** is conceptually similar to time domain envelope but in the frequency domain. It can be seen as the boundary within which the spectrum of a signal is contained. The spectral envelope can be approximated, for example, using linear predictive coding (LPC) [69].
3. **Spectral moments** describe some of the main spectral shape characteristics. They include the spectral centroid, the spectral width, spectral asymmetry and spectral flatness. They are computed in the same way as the temporal waveform moments features by replacing the waveform signal $x[n]$ by the Fourier frequency components $X[t, f]$ of the signal.
4. **Amplitude spectral flatness** is an alternative to the spectral flatness feature. It is computed as the ratio between the geometric and the arithmetic means of the spectral amplitude (globally or in several frequency bands).
5. **Spectral slope** measures the average rate of spectral decrease with frequency (more details can be obtained in Peeters [72]).
6. **Spectral roll-off** is defined as the frequency under which a predefined percentage (typically between 85% and 99%) of the total spectral energy is present.
7. **Spectral flux** characterises the dynamic variation of the spectral information. It is either computed as the derivative of the amplitude spectrum or as the normalised correlation between successive amplitude spectra.
8. **Spectral irregularity features** aims at a finer information description linked to the sound partials (e.g. individual frequency components of a sound). Several approaches have been proposed to estimate these features [72].

In sound scene and event analysis, the temporal and spectral shape features are rarely used separately. In fact, they are mostly simple features designed to model specific aspects of the signal and thus are most often combined with several other

features. The log mel energy features are a notable exception, they are powerful enough to be used on their own as input for classification or feature learning. Only a few earlier studies have compared their individual effectiveness for the task [17, 73]. Instead, the temporal and spectral shape features are more often considered and evaluated together as one set of features sometimes referred to as low-level features.

4.3.3 Cepstral Features

Cepstral features allows the decomposition of the signal according to the so-called source-filter model widely used to model speech production. The signal is then decomposed into a carrier (the source, for speech it can be the glottal excitation) and a modulation (the filter, for speech it includes the vocal tract and the position of the tongue).

1. **Mel frequency cepstral coefficients (MFCC)** are the most common cepstral coefficients [23]. They are obtained as the inverse discrete cosine transform of the log energy in mel frequency bands:

$$\text{mfcc}(t, c) = \sqrt{\frac{2}{M_{\text{mfcc}}}} \sum_{m=1}^{M_{\text{mfcc}}} \log(\tilde{X}_m(t)) \cos\left(\frac{c(m - \frac{1}{2})\pi}{M_{\text{mfcc}}}\right), \quad (4.10)$$

where M_{mfcc} is the number of mel frequency bands, m the frequency band index, $\tilde{X}_m(t)$ is the energy in the m th mel frequency band and c is the index of the cepstrum coefficient ($c \in \{1, 2, \dots, M_{\text{mfcc}}\}$) (see also Fig. 4.4b).

In practice, a common implementation uses a triangular filterbank where each filter is spaced according to a mel frequency scale (4.6) (see also Fig. 4.4a). The energy coefficients $\tilde{X}_m(t)$ in the band m are obtained as a weighted sum of the spectral amplitude components $|X[t, f]|$ (where the weights are given according to the amplitude value of the corresponding triangular filter). The number M_{mfcc} of filters typically varies between 12 and 30 for a bandwidth of 16 kHz. MFCC are widely used for speech processing but they are also among the most popular features for sound scene analysis [73].

2. **Alternative cepstral decompositions** can be obtained similarly to MFCC from other frequency-domain representations. This had led to the introduction of features such as the linear prediction cepstral coefficients (LPCC) based on LPC coefficients, the gammatone feature cepstral coefficients (GFCC) or constant-Q cepstral coefficients (CQCC). None of these features are as popular as the MFCC but GFCC, for example, have been applied to sound scene analysis [74, 90].

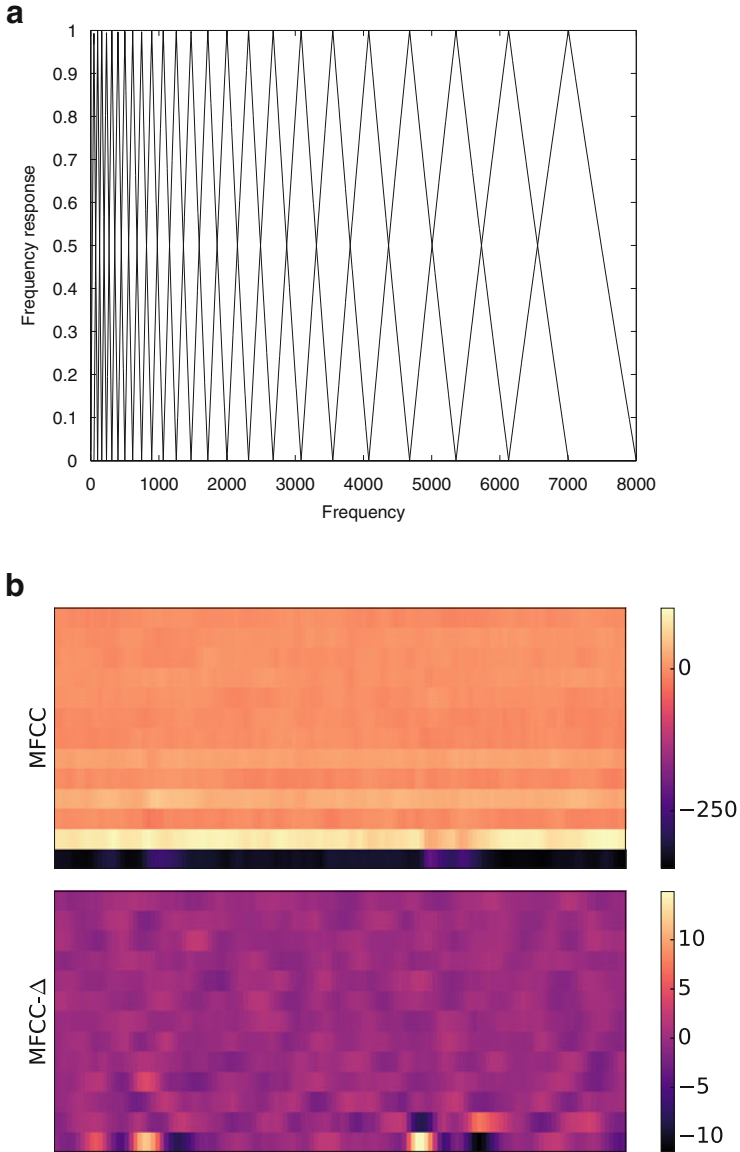


Fig. 4.4 Mel filterbank (a) and MFCC decomposition (b)

4.3.4 *Perceptually Motivated Features*

Studies on human perception have allowed for a better understanding of the human hearing process. Some results from these studies (such as results on auditory filters)

have been exploited in feature engineering and led to widely used features such as MFCC. However, there is still a large variety of perceptual properties that could be exploited in feature extraction (see [80] for a list of common perceptually motivated features for audio classification). To illustrate this category, three perceptual features are described below:

1. **Loudness** (measured in sones) is the subjective impression of the intensity of a sound in such a way that a doubling in sones corresponds to a doubling of loudness. It is commonly obtained as the integration of the specific loudness $L(m)$ over all ERB bands:

$$L = \sum_{m=1}^{M_{\text{ERB}}} L(m) , \quad (4.11)$$

with M_{ERB} the number of ERB bands. The loudness in each band can be approximated [72] by :

$$L(m) = \tilde{X}_m^{0.23} \quad (4.12)$$

where \tilde{X}_m is the energy of the signal in the m th band [see also (4.4)].

2. **Sharpness** can be interpreted as a spectral centroid based on psychoacoustic principle. It is commonly estimated as a weighted centroid of specific loudness [72].
3. **Perceptual spread** is a measure of the timbral width of a given sound. It is computed as the relative difference between the largest specific loudness and the total loudness:

$$S_p = \left(\frac{L - \max_m(L(m))}{L} \right)^2 \quad (4.13)$$

4.3.5 Spectrogram Image-Based Features

Features can also be extracted from the time-frequency representation of a sound scene. Spectrogram image-based features rely on techniques inspired by computer vision to characterise the shape, texture and evolution of the time-frequency content in a sound scene. Such features have proven to be competitive with more traditional audio features on some sound scene classification tasks [47, 78].

1. **Histogram of oriented gradients (HOG)** are image-based features used in computer vision to perform shape detection in images. They are computed from a spectrogram image of a sound scene with the goal of capturing relevant time-frequency structures for characterising sound scenes and events [78]. They are usually extracted by computing a gradient image containing the gradients of each

pixel in a spectrogram image. Each pixel of the gradient image represents the direction of the change in intensity in the original image. After separating the image in non-overlapping cells, a histogram of the gradient orientations for each pixel is computed in each cell. Variations for the HOG features include the choice of the cells, the normalisation of the histograms and the number of orientations.

2. **Subband power distribution (SPD)** relies on a transformation of a time-frequency image into a two-dimensional representation of frequency against spectral power [24]. They are computed by estimating the spectral distribution in each subbands of a spectrogram. In practice the distributions are estimated by extracting a histogram of the pixel values in each subband. The SPD image can directly be used either as features [24] or as an intermediate representation for extracting other image-based features [10].
3. **Local binary pattern (LBP)** analysis is a feature extraction technique used in image recognition to characterise textures in an image. The LBP features are binary vectors associated with each pixel in an image. They are built by comparing the value of a given pixel to others in a fixed neighbourhood. For example, local binary patterns can formed by comparing a given pixel to its eight neighbours, leading to a vector of size eight filled by attributing a value of one to neighbour pixels that have a value above the centre pixel and zero to the others. Similarly to the HOG features, the final LBP features are often obtained by computing the distribution of the different local binary patterns in regions of the image. LBP have been applied sound scene analysis in order to capture the texture and geometrical properties of a scene’s spectrogram [4, 47].

4.3.6 Discussion

“Hand-crafted” features are generally very successful for sound analysis tasks, but very few works in sound scene and event analysis focused on creating features adapted to the specificity of the problem. Instead, a more common approach is to select and adapt features initially introduced for other tasks. A now well-established example of this trend is the popularity of MFCC features in sound scene and event analysis systems. Although many studies have proved the superiority of other “hand-crafted” features for the task, many systems limit themselves to the use of MFCCs while mostly focusing on the classification and detection stage.

One advantage of this approach is that it allows to re-use the work done on MFCC. For example, time domain and frequency-domain filtering [29, 39, 93] to enforce robustness to additive perturbations or cepstral mean normalisation [51] to attenuate the effects of convolutive perturbations. One of the main drawbacks of feature engineering is that it relies on transformations that are defined beforehand and regardless of some particularities of the signals observed at runtime (recording conditions, recording devices, etc.).

4.4 Feature Learning

Representation learning techniques have recently proven superior to manually designed features in many classification and other sound analysis tasks. Indeed more and more datasets of significant size have become available that can be used to develop feature learning techniques. Developments in nonnegative matrix factorisation [52], sparse representation learning [40], dictionary learning [57] and deep learning [8] are manifestations of this trend. This approach allows for extracting features that reflect the underlying structure of the data considered in a particular task, providing high level representations that can generalise, to some extent, to data configurations unseen during the training phase.

The potential of feature learning techniques is particularly clear for sound scene event analysis. In fact, real-life sound events can be of very different nature resulting in a wide variety of possible time-frequency structures present in a sound scene. Moreover, for tasks like sound scene or event classification, only parts of the information are relevant to discriminate the different target sound object classes. The usefulness of feature learning has already been demonstrated on many scene and event classification datasets. For example, works relied on clustering [83], bag-of-features [76, 94] or nonnegative matrix factorisation [5, 11, 65] techniques in order to learn more discriminative representations of sound scenes and events.

4.4.1 *Deep Learning for Feature Extraction*

During the past decade, advances in terms of training algorithms [42, 92] and computing power have led to the generalisation of the use of deep learning techniques [7] that are now the state of the art in many audio applications. Besides their most common application in pattern classification (see also Chap. 5) deep learning techniques such as deep neural networks (DNN) (Chap. 5, Sect. 4.2), convolutional neural networks (CNN) (Chap. 5, Sect. 4.3), recurrent neural networks (RNN) (Chap. 5, Sect. 4.4) can be applied to learn features. A particular type of network architecture that is often used in feature learning are the so-called bottleneck networks (BN) that contain a hidden layer whose size is smaller than other hidden layers. There are then two main different strategies that can be applied to learn features with deep learning:

1. **Supervised learning:** When annotated data is available it is often desired to train the network in a supervised manner in order to learn features that are discriminative between the target classes. At runtime, in the case of DNN, the last hidden layer is used to extract features [41] while in BN it is the bottleneck layer [37] that provides the features.
2. **Unsupervised learning:** With the increasing amount of data at hands it is often the case that at least part of the data available is not annotated. In this case, feature learning will have to rely on unsupervised techniques in order to

extract intrinsic properties of the sound signals. Deep networks can then be trained with restricted Boltzmann machine [42] or stacked autoencoder [92]. In the latter approach, the network is built gradually by combining denoising autoencoders [91]. An autoencoder is a neural network with one hidden layers whose targets are low-level representations of the sound signal. The input of the autoencoder is generally obtained from a (artificially) degraded version of the sound signal. During the training phase the autoencoder then aims at learning how to reconstruct a clean signal from a noisy signal. At runtime, the feature extraction is generally performed similarly as in the supervised case.

More technical details about deep learning in general, network topologies and learning algorithms in particular can be found in Chap. 5, Sect. 5.4.

4.4.2 Matrix Factorisation Techniques

Matrix factorisation (MF) techniques are non-supervised data decomposition techniques, akin to latent variable analysis. In sound analysis applications it generally consists in “explaining” a set of frequency representations for T frames $\{\mathbf{v}_1, \dots, \mathbf{v}_T\}$, as linear combinations of *basis vectors*, also called *dictionary elements*, *atoms*, *elementary patterns* or *topics*. This is accomplished by determining an approximation of the matrix $\mathbf{V} = [v_{f,t}]$ assembled by stacking the observations column-wise, under the form:

$$\mathbf{V} \approx \hat{\mathbf{V}} = \mathbf{W}\mathbf{H} \quad (4.14)$$

where $\mathbf{W} = [w_{fk}]$ is an $F \times K$ -matrix whose columns \mathbf{w}_k are the basis vectors; and $\mathbf{H} = [h_{kt}]$ is a $K \times T$ -matrix whose elements are the so-called *activation coefficients*, *encodings* or *regressors*.

In the following, the t th column of \mathbf{H} will be denoted by \mathbf{h}_t , whereas \mathbf{h}_k : will denote its k th row relating to the sequence of activations of basis vector \mathbf{w}_k .

Generally, MF has been employed as a means of addressing diverse machine learning or signal processing tasks, including clustering, topics recovery, temporal segmentation and structuring, source separation or feature learning. Here, we focus on the latter usage, which have proven effective in sound scene and event analysis applications [12].

In such scenarios, the observations correspond to an appropriate low-level representation, usually a variant of time-frequency representations (described in Sect. 4.2.2), e.g., mel-spectra. These time-frequency representations are analysed by MF, in the training stage, in order to obtain a dictionary \mathbf{W} to be used to decompose both training examples and new test observations \mathbf{v}_t , yielding feature vectors \mathbf{h}_t , to be processed by a classifier.

Various data decomposition methods may actually be described with the matrix factorisation formalism, which optimises different criteria, notably principal com-

ponent analysis (PCA) [43] (see Sect. 4.5.1), independent component analysis [20] and nonnegative matrix factorisation (NMF) [52]. The latter has been found to be a particularly effective feature learning approach in the context of sound scene analysis [11, 21] and event classification [6, 65]. Hence it is briefly described hereafter.

The technique, which has actually been known for more than 30 years, was popularised by Lee et al. [52] who demonstrated its ability to learn “the parts of objects” through an application to face image decomposition. This tendency to decompose data in a “natural” way is due to the constraint imposed to both the dictionary and the activation, that is, all coefficients of \mathbf{W} and \mathbf{H} are constrained to be nonnegative.

\mathbf{W} and \mathbf{H} are obtained by minimising a measure of fit $D(\mathbf{V}|\mathbf{WH})$, while imposing the nonnegativity of \mathbf{W} and \mathbf{H} , which is approached as a constrained optimisation problem. Unfortunately, this problem is not jointly convex in (\mathbf{W}, \mathbf{H}) , and hence admits numerous local and global minima. This is one of the principal reasons that have led researchers to consider imposing different types of additional constraints on \mathbf{W} or \mathbf{H} , based on prior knowledge available when handling a particular application. In many cases, constraints have been expressed through the choice of a form of regularised objective function, such as:

$$C(\mathbf{W}, \mathbf{H}) = D(\mathbf{V}|\mathbf{WH}) + \lambda S(\mathbf{H}) + \eta R(\mathbf{W}) \quad (4.15)$$

where $S(\mathbf{H})$ and $R(\mathbf{W})$ are constraints on the coefficients of \mathbf{H} and \mathbf{W} , respectively. Different types of constraints have been imagined, notably *sparsity constraints*—possibly *group sparsity*—on either \mathbf{W} or \mathbf{H} , which is usually translated into sparsity-inducing penalties (e.g. [26, 44, 87]). Such strategies are quite natural in a feature learning context where they are akin to sparse coding.

Fortunately, for many choices of measure of fit $D(\mathbf{V}|\mathbf{WH})$ and penalties $S(\mathbf{H})$ and $R(\mathbf{W})$, the objective function $C(\mathbf{W}, \mathbf{H})$ is separately convex w.r.t \mathbf{W} for \mathbf{H} fixed and *vice versa*. Consequently, most methods aiming to solve the minimisation problem adopt a *block-coordinate descent* approach whereby update rules are alternately applied to iterates of \mathbf{W} and \mathbf{H} [53].

The choice of an appropriate measure-of-fit function $D(\mathbf{V}|\mathbf{WH})$ is of course crucial. It is usually chosen to be a *separable matrix divergence*, taking the form:

$$D(\mathbf{V}|\hat{\mathbf{V}}) = \sum_{f=1}^K \sum_{t=1}^T d(v_{f,t}|\hat{v}_{ft}) \quad (4.16)$$

where $d(x|y)$ is a scalar divergence. A function $d(x|y)$ is said to be a divergence if it is (1) continuous over x and y ; (2) $d(x|y) \geq 0 \forall x, y \geq 0$ and (3) $d(x|y) = 0$ if and only if $x = y$.

Many variants have been considered in previous works including the β -divergence [28], the general Bregman divergences [25], the α -divergences [19] and Csiszar’s divergences [18], to mention a few of them. When considering sound

signals, it is common to exploit the β -divergence, focusing on particular cases which have proven sufficiently well adapted to our applications. Special cases of the β -divergence yield popular cost functions, namely: the Itakura–Saito (IS) divergence [32] ($\beta = 0$), Kullback–Leibler (KL) divergence ($\beta = 1$) and the ℓ_2 -norm or squared Euclidian distance ($\beta = 2$).

4.4.3 Discussion

Feature learning techniques have seen an increase in popularity for sound scene and event analysis applications in the last few years. They mainly aim at addressing the general limitations of hand-crafted features mentioned in Sect. 4.3.6 and have proven to be viable alternatives. Techniques such as NMF have shown, on multiple occasions, to provide better representations than most feature engineering-based methods. For example, NMF allowed to reach improved performance on sound scene and event classification problems, either by considering the dictionaries learned on individual sounds as features [16] or by keeping the projections on a common dictionary representing the full training data as features [11]. Further improvements have been attained by using sparse and convolutive variants of NMF [11, 21, 48]. Another commonly used dictionary learning technique is probabilistic latent component analysis (a probabilistic equivalent of NMF), which has mostly been applied in its temporally constrained shift-invariant version [5, 6]. Other successful unsupervised feature learning approaches include the use of spherical K-means [83], bag-of-features [76, 94] for classifying sound scenes and events. Interested reader is referred to the corresponding references for further information about these feature learning techniques.

Another trend in sound scene and event analysis has been to introduce supervised variants of some of the feature learning techniques mentioned above. For classification problems, supervised feature learning mainly aims at incorporating prior knowledge about the class labels during the feature learning stage in order to learn more discriminant representations of the data. Once again, several supervised extensions of NMF have been proposed. For acoustic event detection, some works incorporated the sequence of labels in the data before decomposing with NMF or convolutive NMF [48, 65]. Moreover, for sound scene classification, supervision has been introduced to NMF either by learning a nonnegative dictionary and a classifier in a joint optimisation problem [12] or by constraining each dictionary elements to represent only one sound label [77].

4.5 Dimensionality Reduction and Feature Selection

A large number of potentially useful features can be considered in the design of sound scene or event classification systems. Though it is sometimes practicable to use all those features for the classification, it may be sub-optimal to do so,

since many of them may be redundant or, even worse, noisy owing to non-robust extraction procedures. Thus, feature selection or compression (by transformation) become inevitable in order to reduce the complexity of the problem—by reducing its dimensionality—and to retain only the information that is relevant in discriminating the target classes.

4.5.1 Dimensionality Reduction

A common approach to cope with the potentially large dimensionality of the feature space is to use transformation techniques such as PCA, linear discriminant analysis (LDA) or more recent approaches such as the so-called bottleneck DNN. Here, we focus on the popular PCA technique.

PCA, also known as the Karhunen–Loeve transform, computes low-dimensional linear approximations $\hat{\mathbf{v}}$ of the original data points \mathbf{v} in the least-squares sense, that is by seeking a transformation matrix \mathbf{U}^* such that $\mathbf{U}^* = \arg \min_{\mathbf{U}} \|\hat{\mathbf{v}} - \mathbf{v}\|^2$, with $\hat{\mathbf{v}} = \mathbf{U}\mathbf{U}^T\mathbf{v}$ and $\text{rank}(\mathbf{U}) < F$. This can be viewed as a projection of the initial data \mathbf{v} on the new coordinate axes for which the variances of \mathbf{v} on these axes are maximised.

The method is actually a special case of matrix factorisation, previously presented, where $\mathbf{W} = \mathbf{U}$ and $\mathbf{H} = \mathbf{U}^T\mathbf{V}$. Thus, the procedure can be viewed as a projection of the initial data points \mathbf{v} on new coordinate axes, called *principal components*. It is worth noting that other matrix factorisation variants (presented in Sect. 4.4.2) can be used for dimensionality reduction, as long as $K < F$, merely using the activation vectors \mathbf{h}_i as low-dimensional representatives of the original \mathbf{v}_i data points.

Solving the PCA least-squares problem is shown to be equivalent to computing an eigenvalue decomposition (EVD) of the covariance matrix $\mathbf{R}_{\mathbf{v}\mathbf{v}}$ of the data and taking \mathbf{U} to be the K dominant eigenvectors of this decomposition. This yields the best K -dimensional approximation of the original data in the least-squares sense. It can then be easily verified that the covariance matrix of the transformed data is diagonal, hence the components of the transformed data $\hat{\mathbf{v}}$ are uncorrelated, and the first few components (the so-called principal components) capture most of the variance of the original data \mathbf{x} . The interested reader is referred to Murphy [67] for more details about the method.

4.5.2 Feature Selection Paradigms

Feature selection is an interesting alternative to feature transform techniques such as PCA as the latter present the inconvenience of requiring that all candidate features be extracted at the test stage (before the transform found during training is applied to them). Moreover, PCA does not guarantee that noisy features will be

eliminated (since noisy features may exhibit high variance) and the transformed features are difficult to interpret, which is a major drawback if one expects to gain some understanding of the qualities that best describe the classes.

By feature selection (FS), a subset of K' features is selected from a larger set of K candidates with the aim to achieve the lowest classification loss. The task is quite complex: not only is it impracticable to perform the exhaustive subset search because of the extremely high combinatorics involved, as the size of search space is 2^K when K' is not given in advance, but also it is costly to evaluate the classification loss for each candidate feature subset. Therefore feature selection is generally solved in a sub-optimal manner, usually by introducing two main simplifications:

- Brute-force search is avoided by recurring to a near-optimal search strategy.
- Instead of using the classification loss, a simpler feature selection criterion is preferred, which exploits the initial set of features intrinsically, as part of preprocessing stage, before learning the classifiers (using only selected features). This is referred to as *filter approaches* (Sect. 4.5.3), as opposed to the *embedded approaches* (Sect. 4.5.4), where the selection is integrated in the classifier learning process.

4.5.3 Filter Approaches

Such approaches rely on some *subset search method* [54] and *selection criteria*—often heuristic ones, related to class separability (possibly described using a Fisher discriminant), or a measure of the association between features and classes (e.g. mutual information between them).

As for the subset search method, various strategies can be considered [54] which entail choosing a feature subset generation procedure, generally in a sequential way (e.g. forward/backward generation, sequential floating search, random generation, etc.), as well as a sub-optimal search strategy, which may be either deterministic, using heuristics in the choice of the search path (e.g. adding a new feature at a time in a forward generation process), or stochastic (e.g. using simulated annealing or genetic algorithms).

A simpler yet popular approach reduces the task to one of ranking each feature. Here, each individual feature is first scored—independently from the others—using some criterion (say a separability criterion, for example). Then the features are sorted with respect to their scores and the K' top-ranked elements are retained for the classification. Such an approach is clearly sub-optimal compared with the previous search strategies, which does not prevent it from yielding satisfactory performance in practice. Its main advantage is naturally its low complexity.

4.5.4 Embedded Feature Selection

The embedded methods have attracted most of the attention in recent years, taking different forms. Hereafter, we briefly cover the most salient of such approaches.

4.5.4.1 Feature Selection by Sparsity-Inducing Norms

In linear models, including support vector machines (SVM) and generalised linear models [9], feature selection is achieved using a form of regularisation, usually ℓ_1 -norm regularisation in order to promote sparsity of the linear weight vector, as done in the *LASSO* [88]. The classification model estimation problem then takes the general form:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^K} \frac{1}{T} \sum_{t=1}^T \ell(y_t, \boldsymbol{\beta}^\top \mathbf{h}_t) + \alpha \Omega(\boldsymbol{\beta}) ; \quad (4.17)$$

where y_f is the class label associated with feature vector observation \mathbf{h}_t , $\ell(\cdot, \cdot)$ is a classification loss function and $\Omega(\boldsymbol{\beta})$ is a *sparsity-inducing norm*. This norm may be constructed in such a way to account for prior knowledge on the structure of the data, especially to perform *feature-group* selection, as opposed to feature-coefficient selection [3, 45]. Such a selection process (aka *feature-subset selection*) may be more advantageous, since it may be known in advance that some variables do not make sense when isolated from a “natural” group to which they belong. Moreover, this may allow for implicitly selecting only a subset of channels, in multi-channel setups (again provided that different feature groups are associated with them) which in practice is very valuable, as this could result in a simplification of the hardware used for capturing the data.

4.5.4.2 Multiple Kernel Learning

A set of advanced feature selection techniques have been developed for kernel-based methods [84], especially SVM classifiers, within the framework of *multiple kernel learning* (MKL) [50, 79, 85]. Here the main principle is to learn the kernel κ_0 to be used by the classifier as a convex combination of predefined base kernels κ_r according to: $\kappa_0(\mathbf{h}, \mathbf{h}') = \sum_{r=1}^R \mu_r \kappa_r(\mathbf{h}, \mathbf{h}')$. Now by defining the different base kernels on different feature groups (possibly different feature coefficients in the extreme case), and with a proper formulation of the classifier learning problem, involving sparsity-promoting penalties [79], only a subset of the considered kernels will have non-zero weights in the final solution, hence only a subset of features will be retained.

4.5.4.3 Feature Selection in Tree-Based Classifiers

In classification schemes based on trees, possibly under *boosting* or *random-forest* settings [38, Chap. 10], feature selection often comes as a by-product of the classifier learning process, which may occur at various levels: either at the stage of the actual tree growing process, where at each node a particular feature (or feature set) is naturally selected; or at the level of the *ensemble* meta-classifier, which through the selection of the weak classifiers (in boosting schemes) or the random sub-sampling of the variables (in random forests), retains at the end of the learning only the subset of the most useful features. Additionally, further dimensionality reduction can be accomplished as part of a post-processing stage where efficient procedures for *variable importance determination* and *pruning* exist [38, Chap. 10].

4.6 Temporal Integration and Pooling

Most of the features described above (see Sect. 4.3) capture specific properties of the given signal over short-time signal analysis windows (or *frames*) over which the signal can be considered stationary. Then, it is commonly assumed that the successive observations of features in different frames are statistically independent, which means that the time evolution of these features is neglected for classification. In this section, we describe several strategies, often termed *temporal integration*, to take into account the information conveyed in the temporal evolution of the signal.

4.6.1 Temporal Integration by Simple Statistics

Temporal integration can be directly performed on the “instantaneous” features computed locally over short analysis frames. This so-called early integration is then commonly done over larger time windows called *texture windows* (see Fig. 4.5). The early temporal integration process can be represented by a function g which is applied on a sequence of feature vectors, noted $\mathbf{h}_t = [h_{1,t} \ h_{2,t} \ \dots \ h_{K,t}]$ where $h_{f,t}$ corresponds to the f th scalar feature observed in the t th frame.

The aims of the integration function is to either capture short-time statistics (such as the mean and covariance described below) or to more complex temporal integration using some kind of models (see Sect. 4.6.2). A straightforward mean for early integration is to compute first order statistics of the feature process. The *mean* integration function is then defined as

$$g_{\text{mean}}(\mathbf{h}_t, \dots, \mathbf{h}_{t+N-1}) = \mu_t = \frac{1}{N} \sum_{k=t}^{t+N-1} \mathbf{h}_k . \quad (4.18)$$

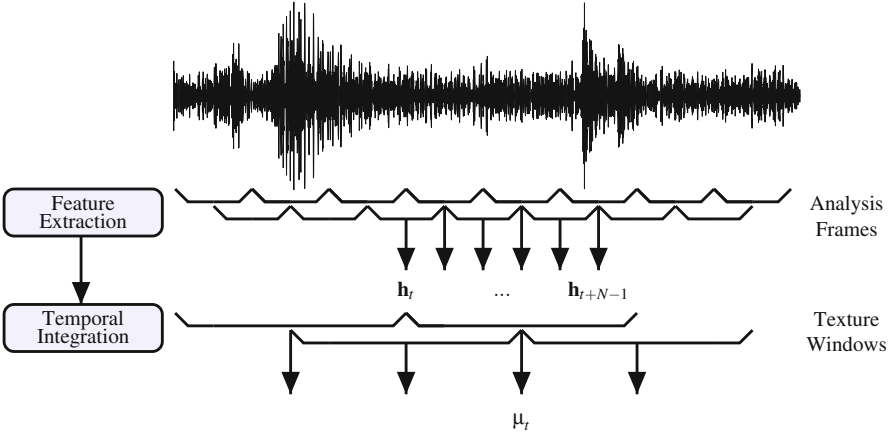


Fig. 4.5 Illustration of the different windows used (analysis frame and texture window)

This simple approach can be extended max-abs pooling (Chap. 5, Sect. 4.3) or to higher order statistics using, for example, the full covariance matrix (or only the empirical variance of the features), the skewness or kurtosis (see, for example, [46, 61, 89] for some examples on music signal processing applications).

4.6.2 Model-Based Integration

More sophisticated models can also be used to model the temporal dependency between successive features. It is, for example, possible to model the sequence of features as an autoregressive process. Such a model will capture some global spectral properties, where the level of details depends on the order of the AR model. Following the multivariate autoregressive model used in Meng [63] for music genre classification, the corresponding integration function g_{MAR} can be written as :

$$g_{\text{MAR}}(\mathbf{h}_t, \dots, \mathbf{h}_{t+N-1}) = \left[\hat{\mathbf{w}} \hat{\mathbf{A}}_1 \dots \hat{\mathbf{A}}_p \right], \quad (4.19)$$

where $\hat{\mathbf{w}}$ and $\{\hat{\mathbf{A}}_p\}_{p=1, \dots, P}$ are the least-square estimators of the model parameters for the t texture window and where the p th order model, denoted by $\text{MAR}(p)$ is defined as:

$$\mathbf{h}_t = \hat{\mathbf{w}} + \sum_{p=1}^P \mathbf{h}_{t-p} \hat{\mathbf{A}}_p + \boldsymbol{\varepsilon}_t, \quad (4.20)$$

with $\boldsymbol{\varepsilon}_t$ being a D -dimensional white noise vector.

A number of variations of this model have been proposed including, for example, the diagonal autoregressive model or the centred autoregressive model.

Direct extensions of the previous concepts aim at computing spectral characteristics of the feature sequence. Such integrated features include, for example, the modulation energy of “instantaneous” MFCC features [62], the spectral moments of the feature sequence over a texture window, the STFT coefficients (or as more recently proposed the coefficients of the scattering transform) for every feature over a texture window.

An alternative strategy will consist in incorporating some learning or classification paradigms in the feature calculation. It is, for example, possible to estimate the probability density of the feature sequence over a texture window and to model it using a Gaussian mixture model (GMM), GMM super-vectors or even I-vectors [27]. Since these integration approaches can be considered as part of the classification algorithm, they are not further discussed herein.

4.6.3 Discussion

In sound scene and event analysis, the importance accorded to temporal integration of features largely depends on the target problem. First, for a task like sound event detection, where precise estimation of event onset times is required, the use of temporal integration is rather uncommon. Instead, the temporal information of the sound scene is modelled during the classification stage by using techniques such as hidden Markov models [30, 64], RNN [1, 70] or CNN for finite context [75].

The importance of temporal integration is particularly clear for other tasks like sound scene and event classification, where the decision is taken on longer segments of sound. Because of the frame-based nature of many of these features, a particular focus on temporal integration is required in order to model the distribution of the features across the full duration of the sound examples. In that case, the most common approaches are either to classify the frame-based features before performing voting strategies or to directly classify statistics of frame-based features computed over the full duration of the sound (see also late and early fusion techniques in Chap. 5). In the latter case, the most common way of modelling the temporal information is either to extend the feature set with their first and second order derivatives or to compute their average over time, possibly combined with more complex statistical functions [34, 49]. The use of *Recursive Quantitative Analysis* [81] on frame-based features has also proven to be effective for modelling temporal information.

4.7 Relation to Work on Speech and Music Processing

Speech and music are specific examples of sound signals and, as such, share many acoustical characteristics with sound scenes and sound events recordings. Speech processing is a well-established field with a long history of research. Numerous features have then been proposed to characterise speech signals and used in several major classification tasks such as speech recognition or speaker identification. Music signal processing, although more recent than speech, is nevertheless another major domain of audio signal processing with a strong history.

It is therefore not surprising that a large body of features formerly introduced in speech and music research has been directly applied to sound scene or sound event recognition. ZCR, filterbanks, cepstral features and a number of perceptually motivated features [80] were indeed proposed previously for varied sound classification tasks.

In particular, the MFCC described in Sect. 4.3.3, remain, even today, one of the most widely used features in sound classification since its initial use for a music processing task by Logan [55]. This is surprising since MFCC were initially designed for processing speech signals and in particular for speech recognition [23]. In fact, MFCC integrate some perception properties and, with reference to the classic speech source-filter production model, mainly discard the source part making the MFCC rather pitch independent. A direct application of MFCC for music and environmental sound analysis is surprising since (1) the pitch range is much wider in general sound signals than in speech; (2) for high pitches the deconvolution property of MFCCs does not hold anymore (e.g. MFCC become pitch dependent) and (3) MFCC are not highly correlated with the perceptual dimensions of “polyphonic timbre” in music signals despite their widespread use as predictors of perceived similarity of timbre [2, 64, 80]. It seems, however, that the MFCC’s capacity to capture “global” spectral envelope properties is the main reason of their success in sound classification tasks.

However, it is worth emphasising that some recent proposals targeted features especially designed for sound scenes or sound events recognition. These include, for example, the matching pursuit-based features proposed in Chu et al. [17], the image-based histogram features proposed in Rakotomamonjy et al. [78] or the learned matrix factorisation features [11]. Indeed, the problem of sound scene and sound event recognition is different and calls for features that are adapted to the specificities of the problem, to the scarcity of training (annotated) data and to the fact that individual classes (especially events) may be only observed in mixtures.

4.8 Conclusion and Future Directions

In this chapter we have presented an overview of the different blocks of a standard feature extraction process. The analysis of sound scene and events is a relatively new field of research in the context of sound signal analysis in general. Thus, the majority

of the techniques presented in this chapter were introduced for other applications and have only later been applied to address sound scene analysis problems. In fact, many early works focused on comparing the effectiveness of different previously existing feature extraction techniques with strong inspirations from speech and music processing techniques.

We have shown that the first step in most feature extraction techniques is the choice of a suited time-frequency representation. Being at the beginning of the processing chain they play crucial role in building a sound scene analysis system. However, the choice of the representation and its parameters is rarely justified apart from stating the perceptually motivated aspect of most of them. As mentioned, many systems directly input such representations into the classification stage especially for deep learning techniques. Therefore, the performance of such systems can be limited to the quality of the representation/features used for training. Hence, the sound scene and event analysis field would benefit more in-depth studies of the advantages and drawbacks of certain representation to accurately describe and discriminate the useful information in sound scenes. Moreover, new alternative representations have emerged, mostly based on scattering transforms, and have provided significant increases in performance for some problems.

We have also presented a selection of the most frequently used hand-crafted features. It is still common to see the introduction of new features for sound scene and event analysis mainly inspired from speech, music or image processing. The study of hand-crafted features often brings interesting insight on the content and behaviour of sound scenes. However, they are often limited to describing only specific aspects of the time-frequency information. Multiple studies have exhibited this limitation of hand-crafted features by showing that combining a large variety of different features is often required to improve performance over features taken in isolation.

Finally, the most recent performance breakthroughs in sound scene and event analysis have been attained by using feature learning based on MF or deep neural network techniques. These have the advantage of automatically learning the relevant information in the data often directly from time-frequency representations. Therefore they allow for bypassing the exploration and engineering effort of choosing suited features for the task. However, deep learning techniques require their own kind of engineering effort for finding the appropriate architecture for the target task, which is highly dependent on the content and size of the datasets. In contrary, MF techniques for feature learning demand a lot less tuning effort and have shown on many occasions to be competitive with deep learning systems even when using simple classifiers. We believe that future progress in the field will be highly conditioned on the release of new larger datasets, which will further increase the effectiveness of deep learning techniques, as well as future developments in unsupervised or supervised feature learning techniques such as matrix factorisation.

References

1. Adavanne, S., Parascandolo, G., Pertila, P., Heittola, T., Virtanen, T.: Sound event detection in multichannel audio using spatial and harmonic features. In: Proceedings of Workshop on Detection and Classification of Acoustic Scenes Events, pp. 6–10 (2016)
2. Alluri, V., Toiviainen, P.: Exploring perceptual and acoustical correlates of polyphonic timbre. *Music. Percept.* **27**(3), 223–241 (2010)
3. Bach, F.R., Jenatton, R., Mairal, J., Obozinski, G.: Optimization with sparsity-inducing penalties. Technical Report, INRIA - SIERRA Project Team (2011)
4. Battaglino, D., Lepauloux, L., Pilati, L., Evansi, N.: Acoustic context recognition using local binary pattern codebooks. In: Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 1–5 (2015)
5. Benetos, E., Lagrange, M., Dixon, S.: Characterisation of acoustic scenes using a temporally constrained shift-invariant model. In: Proceedings of Conference on Digital Audio Effects (2012)
6. Benetos, E., Lagrange, M., Plumbley, M.D., et al.: Detection of overlapping acoustic events using a temporally-constrained probabilistic model. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6450–6454. IEEE, New York (2016)
7. Bengio, Y.: Learning deep architectures for AI. *Found. Trends Mach. Learn.* **2**(1), 1–127 (2009)
8. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(8), 1798–1828 (2013)
9. Bishop, C.M.: *Pattern Recognition and Machine Learning*. Springer, Berlin (2006)
10. Bisot, V., Essid, S., Richard, G.: HOG and subband power distribution image features for acoustic scene classification. In: Proceedings of European Signal Processing Conference, pp. 719–723 (2015)
11. Bisot, V., Serizel, R., Essid, S., Richard, G.: Acoustic scene classification with matrix factorization for unsupervised feature learning. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6445–6449 (2016)
12. Bisot, V., Serizel, R., Essid, S., Richard, G.: Feature learning with matrix factorization applied to acoustic scene classification. *IEEE Trans. Audio Speech Lang. Process.* **25**(6), 1216–1229 (2017)
13. Brown, J.C.: Calculation of a constant Q spectral transform. *J. Acoust. Soc. Am.* **89**(1), 425–434 (1991)
14. Burt, P.J.: Fast filter transform for image processing. *Comput. Graphics Image Process.* **16**(1), 20–51 (1981)
15. Burt, P., Adelson, E.: The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.* **31**(4), 532–540 (1983)
16. Cauchi, B.: Non-negative matrix factorization applied to auditory scene classification. Master’s Thesis, ATIAM (UPMC/IRCAM/TELECOM ParisTech) (2011)
17. Chu, S., Narayanan, S., Kuo, C.C.J.: Environmental sound recognition with time-frequency audio features. *IEEE Trans. Audio Speech Lang. Process.* **17**(6), 1142–1158 (2009)
18. Cichocki, A., Zdunek, R., Amari, S.: Csiszar’s divergences for non-negative matrix factorization: family of new algorithms. In: Proceedings of International Conference on Independent Component Analysis and Blind Signal Separation, Charleston, SC, pp. 32–39 (2006)
19. Cichocki, A., Lee, H., Kim, Y.D., Choi, S.: Non-negative matrix factorization with α -divergence. *Pattern Recogn. Lett.* **29**(9), 1433–1440 (2008)
20. Comon, P., Jutten, C.: *Handbook of Blind Source Separation, Independent Component Analysis and Applications*. Academic Press/Elsevier, Amsterdam (2010)
21. Cotton, C.V., Ellis, D.: Spectral vs. spectro-temporal features for acoustic event detection. In: Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 69–72. IEEE, New York (2011)

22. Croisier, A., Esteban, D., Galand, C.: Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. In: International Conference on Information Science and Systems, Patras, vol. 2, pp. 443–446 (1976)
23. Davis, S., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech Signal Process.* **28**(4), 357–366 (1980)
24. Dennis, J., Tran, H.D., Chng, E.S.: Image feature representation of the subband power distribution for robust sound event classification. *IEEE Trans. Audio Speech Lang. Process.* **21**(2), 367–377 (2013)
25. Dhillon, I., Sra, S.: Generalized nonnegative matrix approximations with Bregman divergences. In: Proceedings of Advances in Neural Information Processing Systems, vol. 19, pp. 290, 283 (2005)
26. Eggert, J., Korner, E.: Sparse coding and NMF. In: Proceedings of the International Joint Conference on Neural Networks, vol. 4, pp. 2529–2533. IEEE, New York (2004)
27. Eghbal-Zadeh, H., Lehner, B., Dorfer, M., Widmer, G.: CP-JKU submissions for DCASE-2016: a hybrid approach using binaural i-vectors and deep convolutional neural networks. Technical Report, DCASE2016 Challenge (2016)
28. Eguchi, S., Kano, Y.: Robustifying maximum likelihood estimation. Technical Report, Institute of Statistical Mathematics (2001)
29. Ephraim, Y., Malah, D.: Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE Trans. Acoust. Speech Signal Process.* **33**(2), 443–445 (1985)
30. Eronen, A.J., Peltonen, V.T., Tuomi, J.T., Klapuri, A.P., Fagerlund, S., Sorsa, T., Lorho, G., Huopaniemi, J.: Audio-based context recognition. *IEEE Trans. Audio Speech Lang. Process.* **14**(1), 321–329 (2006)
31. Fant, G.: Analysis and synthesis of speech processes. In: Malmberg, B. (ed.) *Manual of Phonetics*, chap. 8, pp. 173–277. North-Holland, Amsterdam (1968)
32. Févotte, C., Bertin, N., Durrieu, J.L.: Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis. *Neural Comput.* **21**(3) (2009)
33. Fletcher, H.: Auditory patterns. *Rev. Mod. Phys.* **12**(1), 47 (1940)
34. Geiger, J.T., Schuller, B., Rigoll, G.: Large-scale audio feature extraction and SVM for acoustic scene classification. In: Proceeding of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (2013)
35. Glasberg, B.R., Moore, B.C.: Derivation of auditory filter shapes from notched-noise data. *Hear. Res.* **47**(1–2), 103–138 (1990)
36. Goupillaud, P., Grossmann, A., Morlet, J.: Cycle-octave and related transforms in seismic signal analysis. *Geoexploration* **23**(1), 85–102 (1984)
37. Grézl, F., Karafiát, M., Kontár, S., Cernocký, J.: Probabilistic and bottle-neck features for LVCSR of meetings. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 4, p. 757. IEEE, New York (2007)
38. Hastie, T., Tibshirani, R., Friedman, J.H.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York (2009)
39. Haykin, S.: *Adaptive Filter Theory*, 5th edn. Pearson Education, Upper Saddle River (2014)
40. Henaff, M., Jarrett, K., Kavukcuoglu, K., LeCun, Y.: Unsupervised learning of sparse features for scalable audio classification. In: Proceedings of International Society for Music Information Retrieval Conference, vol. 11, p. 2011 (2011)
41. Hermansky, H., Ellis, D.P., Sharma, S.: Tandem connectionist feature extraction for conventional HMM systems. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 3, pp. 1635–1638. IEEE, New York (2000)
42. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. *Neural Comput.* **18**(7), 1527–1554 (2006)
43. Hotelling, H.: Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* **24**, 417–441 (1933)
44. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.* **5**, 1457–1469 (2004)

45. Jenatton, R., Audibert, J.Y., Bach, F.: Structured variable selection with sparsity-inducing norms. *J. Mach. Learn. Res.* **12**, 2777–2824 (2009). arXiv:0904.3523
46. Joder, C., Essid, S., Richard, G.: Temporal integration for audio classification with application to musical instrument classification. *IEEE Trans. Audio Speech Lang. Process.* **17**(1), 174–186 (2009)
47. Kobayashi, T., Ye, J.: Acoustic feature extraction by statistics based local binary pattern for environmental sound classification. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3052–3056. IEEE, New York (2014)
48. Komatsu, T., Toizumi, T., Kondo, R., Senda, Y.: Acoustic event detection method using semi-supervised non-negative matrix factorization with mixtures of local dictionaries. In: *Proceedings of IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events*, pp. 45–49 (2016)
49. Krijnders, J., Holt, G.A.T.: A tone-fit feature representation for scene classification. In: *Proceedings of IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events* (2013)
50. Lanckriet, G.R.G., Bartlett, P., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.* **5**, 27–72 (2004)
51. Lee, L., Rose, R.C.: Speaker normalization using efficient frequency warping procedures. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pp. 353–356. IEEE, New York (1996)
52. Lee, L., Seung, S.: Learning the parts of objects with nonnegative matrix factorization. *Nature* **401**, 788–791 (1999)
53. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: *Proceedings of Advances in Neural Information Processing Systems*, pp. 556–562 (2001)
54. Liu, H., Motoda, H.: *Feature Selection for Knowledge Discovery and Data Mining*, 2nd edn. Kluwer Academic, New York (2000)
55. Logan, B.: Mel frequency cepstral coefficients for music modeling. In: *Proceedings of International Society for Music Information Retrieval Conference* (2000)
56. Lostanlen, V., Andén, J.: Binaural scene classification with wavelet scattering. Technical Report, DCASE2016 Challenge (2016)
57. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: *Proceedings of International Conference on Machine Learning*, pp. 689–696. ACM, New York (2009)
58. Mallat, S.G.: Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process.* **37**(12), 2091–2110 (1989)
59. Mallat, S.: *A Wavelet Tour of Signal Processing*. Academic Press, London (1999)
60. Mallat, S.: Group invariant scattering. *Commun. Pure Appl. Math.* **65**(10), 1331–1398 (2012)
61. Mandel, M., Ellis, D.: Song-level features and SVMs for music classification. In: *Proceedings of International Society for Music Information Retrieval Conference* (2005)
62. McKinney, M.F., Breebart, J.: Features for audio and music classification. In: *International Symposium on Music Information Retrieval*, pp. 151–158 (2003)
63. Meng, A.: Temporal feature integration for music organisation. Ph.D. Thesis, Technical University of Denmark (2006)
64. Mesaros, A., Virtanen, T.: Automatic recognition of lyrics in singing. *EURASIP J. Audio Speech Music Process.* **2010**(1), 546,047 (2010)
65. Mesaros, A., Heittola, T., Dikmen, O., Virtanen, T.: Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 151–155 (2015)
66. Mesaros, A., Heittola, T., Virtanen, T.: TUT database for acoustic scene classification and sound event detection. In: *Proceedings of European Signal Processing Conference* (2016)
67. Murphy, K.P.: *Machine Learning: A Probabilistic Perspective*. MIT Press, Cambridge (2012)
68. O’Shaughnessy, D.: *Speech Communication: Human and Machine*. Addison-Wesley Series in Electrical Engineering. Addison-Wesley, Boston (1987)

69. O'Shaughnessy, D.: Linear predictive coding. *IEEE Potentials* **7**(1), 29–32 (1988)
70. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6440–6444. IEEE, New York (2016)
71. Patterson, R.D., Robinson, K., Holdsworth, J., McKeown, D., Zhang, C., Allerhand, M.: Complex sounds and auditory images. In: *Proceedings of International Symposium on Hearing*, vol. 83, pp. 429–446. Pergamon, Oxford (1992)
72. Peeters, G.: A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Technical Report, IRCAM, Paris (2004)
73. Peltonen, V., Tuomi, J., Klapuri, A., Huopaniemi, J., Sorsa, T.: Computational auditory scene recognition. In: *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, vol. 2, p. 1941 (2002)
74. Phan, H., Hertel, L., Maass, M., Koch, P., Mertins, A.: Car-forest: joint classification-regression decision forests for overlapping audio event detection (DCASE). Technical Report, DCASE2016 Challenge (2016)
75. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: *Proceedings of International Workshop on Machine Learning for Signal Processing*, pp. 1–6. IEEE, New York (2015)
76. Plinge, A., Grzeszick, R., Fink, G.A.: A bag-of-features approach to acoustic event detection. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3704–3708. IEEE, New York (2014)
77. Rakotomamonjy, A.: Enriched supervised feature learning for acoustic scene classification. Technical Report, DCASE2016 Challenge (2016)
78. Rakotomamonjy, A., Gasso, G.: Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(1), 142–153 (2015)
79. Rakotomamonjy, A., Bach, F.R., Canu, S., Grandvalet, Y.: SimpleMKL. *J. Mach. Learn. Res.* **9**, 2491–2521 (2008)
80. Richard, G., Sundaram, S., Narayanan, S.: An overview on perceptually motivated audio indexing and classification. *Proc. IEEE* **101**(9), 1939–1954 (2013)
81. Roma, G., Nogueira, W., Herrera, P.: Recurrence quantification analysis features for environmental sound recognition. In: *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (2013)
82. Salamon, J., Bello, J.B.: Feature learning with deep scattering for urban sound analysis. In: *Proceedings of European Signal Processing Conference*, pp. 724–728. IEEE, New York (2015)
83. Salamon, J., Bello, J.P.: Unsupervised feature learning for urban sound classification. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 171–175 (2015)
84. Shölkopf, B., Smola, A.J.: *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge (2002)
85. Sonnenburg, S., Rätsch, G., Schäfer, C., Schölkopf, B.: Large scale multiple kernel learning. *J. Mach. Learn. Res.* **7**, 1531–1565 (2006)
86. Stevens, S.S., Volkman, J., Newman, E.B.: A scale for the measurement of the psychological magnitude pitch. *J. Acoust. Soc. Am.* **8**(3), 185–190 (1937)
87. Sun, D., Mazumder, R.: Non-negative matrix completion for bandwidth extension: a convex optimization approach. In: *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing* (2013)
88. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **58**, 267–288 (1996)
89. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. In: *IEEE Transactions on Speech and Audio Processing* (2002)
90. Valero, X., Alías, F.: Gammatone cepstral coefficients: biologically inspired features for non-speech audio classification. *IEEE Trans. Multimedia* **14**(6), 1684–1689 (2012)

91. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the International Conference on Machine Learning, pp. 1096–1103. ACM, New York (2008)
92. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **11**, 3371–3408 (2010)
93. Widrow, B., Stearns, S.D.: Adaptive Signal Processing, 1st edn., p. 491. Prentice-Hall, Englewood Cliffs (1985)
94. Ye, J., Kobayashi, T., Murakawa, M., Higuchi, T.: Acoustic scene classification based on sound textures and events. In: Proceedings of Annual Conference on Multimedia, pp. 1291–1294 (2015)
95. Zwicker, E., Terhardt, E.: Analytical expressions for critical-band rate and critical bandwidth as a function of frequency. *J. Acoust. Soc. Am.* **68**(5), 1523–1525 (1980)

Chapter 5

Statistical Methods for Scene and Event Classification

Brian McFee

Abstract This chapter surveys methods for pattern classification in audio data. Broadly speaking, these methods take as input some representation of audio, typically the raw waveform or a time-frequency spectrogram, and produce semantically meaningful classification of its contents. We begin with a brief overview of statistical modeling, supervised machine learning, and model validation. This is followed by a survey of discriminative models for binary and multi-class classification problems. Next, we provide an overview of generative probabilistic models, including both maximum likelihood and Bayesian parameter estimation. We focus specifically on Gaussian mixture models and hidden Markov models, and their application to audio and time-series data. We then describe modern deep learning architectures, including convolutional networks, different variants of recurrent neural networks, and hybrid models. Finally, we survey model-agnostic techniques for improving the stability of classifiers.

Keywords Machine learning • Statistical modeling • Classification • Discriminative models • Generative models • Deep learning • Convolutional neural networks • Recurrent neural networks • Hidden Markov models • Bayesian inference

5.1 Introduction

This chapter provides an overview of machine learning methods for pattern classification. Throughout this chapter, our objective is to design algorithms which take as input some representation of an audio signal, and produce some semantically meaningful output, e.g., a categorical label indicating the presence of an acoustic event in the audio signal.

The treatment of topics in this chapter will be relatively superficial: our goal is to provide a high-level overview of methods for pattern classification, not an in-depth

B. McFee (✉)
Center for Data Science, New York University, 60 5th Ave., New York, NY 10003, USA
e-mail: brian.mcfee@nyu.edu

survey of advanced statistics and machine learning. We will assume familiarity with linear algebra, multivariate calculus, and elementary probability theory and statistics. We will not cover computational learning theory or optimization, but references for those concepts will be provided.

The remainder of this chapter is structured as follows. Section 5.1 describes the fundamentals and practical considerations of statistical learning. Section 5.2 introduces discriminative models for binary and multi-class prediction problems, with a focus on linear models. Section 5.3 covers generative models, unsupervised learning, and Bayesian inference, focusing on Gaussian mixture models and hidden Markov models for audio applications. Section 5.4 provides an overview of deep learning, including multi-layer perceptrons, one- and two-dimensional convolutional networks, various formulations of recurrent neural networks, and hybrid architectures. Section 5.5 describes some useful techniques to improve the robustness and stability of classifiers. Finally, Sect. 5.6 concludes with pointers to further readings on advanced topics.

Throughout this chapter, the input representation of audio is generally left abstract, and may correspond to a summary of an entire recording or more localized representations of individual audio frames. The fundamentals of binary and multi-class discriminative classifiers described in Sect. 5.2 apply to both of these cases. For example, a static acoustic scene classification system could apply a multi-class discriminative classifier to a feature vector representing the acoustic properties of the entire audio recording, resulting in a single categorical label predicted for the entire recording. Similarly, a clip-level tagging system could apply several binary classifiers to predict the presence of multiple concepts within a recording (e.g., *speech*, *bird song*, *footsteps*), but without localizing them in time. By contrast, *dynamic* prediction tasks, such as sound event detection, would operate on localized representations (e.g., individual frames) to produce a time-series of predictions. Methods for exploiting temporal structure are described in Sect. 5.3.5 (Hidden Markov models) and Sects. 5.4.3 and 5.4.4 (convolutional and recurrent networks).

5.1.1 Preliminaries

Input data will be generically denoted as $x \in \mathcal{X}$, and output variables will be denoted as $y \in \mathcal{Y}$. The input domain \mathcal{X} and output space \mathcal{Y} will be left abstract for much of this chapter, but it may be helpful to think of the concrete case where $\mathcal{X} = \mathbb{R}^d$ corresponds to some pre-computed frame-level features (e.g., mel-scaled power spectra as described in Chap. 4) and $\mathcal{Y} = \{-1, +1\}$ are binary categorical labels. Input–output pairs are assumed to be jointly distributed according to some (unknown) probability distribution $(x, y) \sim \mathcal{D}$; for brevity, we will sometimes write $z = (x, y)$ to denote a labeled example. A classifier (or, more generally, a predictor) will map an observed input x to an output (label) y , and be denoted as $h : \mathcal{X} \rightarrow \mathcal{Y}$. Finally, we will characterize the accuracy of a predictor by using *loss functions*, denoted by $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+$, to compare an estimated label $h(x)$ to a true label y . Small values of ℓ indicate high accuracy, and high values of ℓ indicate low accuracy.

This chapter is primarily concerned with the *supervised learning* model, wherein a sample of labeled points $S = \{(x_i, y_i)\}_{i=1}^n$ (the *training set*) are independently and identically distributed (*I.I.D.*) by a probability distribution \mathcal{D} , and used to estimate the parameters of the algorithm. In general, we would like to find a predictor h that minimizes the *risk*¹:

$$\mathbf{E}_{\mathcal{D}} [\ell(h(x), y)] = \int_{x,y} \ell(h(x), y) \times \mathbf{P}_{\mathcal{D}}[x, y] dx dy. \quad (5.1)$$

Put plainly, (5.1) captures the expected error rate of a predictor h over the data distribution \mathcal{D} . When ℓ is the 0–1 loss:

$$\ell(y, y') := \begin{cases} 0 & y = y' \\ 1 & y \neq y' \end{cases} \quad (5.2)$$

then (5.1) is the probability of incorrectly classifying a randomly selected input x . Since \mathcal{D} is generally unknown, minimizing (5.1) over choices of h is not possible. The supervised learning approach is to approximate (5.1) by the *empirical risk* estimated over the sample:

$$\frac{1}{n} \sum_{i=1}^n \ell(h(x_i), y_i) \approx \mathbf{E}_{\mathcal{D}} [\ell(h(x), y)]. \quad (5.3)$$

The learning problem therefore amounts to minimizing an objective function (5.3) to solve for h over some class of models.

The predictor h is generally defined in terms of parameters $\theta \in \Theta$, which we denote as $h(x | \theta)$. Thus, the learning problem can be generally stated as minimizing (5.3) over the choice of θ from a space Θ of possible configurations:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n \ell(h(x_i | \theta), y_i). \quad (5.4)$$

When ℓ is continuous and differentiable—such as in least-squares regression, where $\ell(y, y') = \|y - y'\|^2$ —then (5.4) can be solved by iterative methods such as gradient descent, or occasionally in closed form. However, for classification problems, ℓ is often discontinuous or non-differentiable; for example, the 0–1 loss (5.2) is neither continuous nor differentiable with respect to θ . In these cases, exactly optimizing (5.4) can be a difficult computational problem [45, 74]. As a result, it is common to replace the exact loss function ℓ with a *surrogate function* f that is amenable to efficient optimization: typically this means that f is continuous and (at least piece-wise) differentiable.

¹The notation $\mathbf{P}_{\mathcal{D}}$ denotes the probability mass (or density) with respect to distribution \mathcal{D} , and $\mathbf{E}_{\mathcal{D}}$ denotes the expectation with respect to distribution \mathcal{D} .

Surrogate objective functions may operate not directly upon the predicted label $h(x | \theta)$, but on some convenient, related quantity such as conditional probability of a category given the observed x . In general, we will denote the surrogate loss as a function $f : \mathcal{X} \times \mathcal{Y} \times \Theta \rightarrow \mathbb{R}_+$. To summarize, this chain of steps leads to a general formulation of learning:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n f(x_i, y_i | \theta), \quad (5.5)$$

where minimizing (5.5) approximately minimizes (5.4), which in turn approximates the risk (5.3) which we would ideally minimize.²

Finally, one may wish to encode some preferences for certain configurations of θ over others. This can be achieved by including a *regularization function* or *penalty term* $g : \Theta \rightarrow \mathbb{R}_+$ which takes low values for preferred configurations and high values for undesirable configurations. The regularized learning objective then takes the general form we will use for the remainder of this chapter:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n f(x_i, y_i | \theta) + g(\theta). \quad (5.6)$$

As we will see in Sect. 5.3, the general form of (5.6) can also be used for maximum a posteriori Bayesian inference, where the g term takes the role of the prior distribution on the model parameters.

5.1.2 Validation and Testing

The previous section outlines a generic recipe for building predictive models:

1. Collect a labeled training sample S ,
2. Specify a surrogate loss function f and penalty g ,
3. Solve (5.6) to find parameters θ^* ,
4. Deploy the resulting model $h(\cdot | \theta^*)$.

In practice, before deploying the model θ^* , we would also like to have an estimate of how well it performs on unseen data drawn from \mathcal{D} . This can be estimated by using a second independent sample $S_T \sim \mathcal{D}$ known as the *test set*, which is only used for evaluating θ^* and not for parameter estimation.

²Quantifying the relationships between (5.5), (5.4), and (5.3) lies within the purview of statistics and computational learning theory, and is beyond the scope of this text. We refer interested readers to [48, 106] for an introduction to the subject.

By the same token, it is common for practitioners to develop multiple candidate models, generally derived from different choices of (f, g) . Before moving on to testing and deployment, the practitioner must choose a particular model from among the candidates. This process is commonly referred to as *validation* or *hyper-parameter optimization*. It is important to note that the test set S_T cannot be used for validation. Any sample data which influences the selection of θ must be interpreted as *training data*, regardless of whether it appears in (5.6).

The typical approach to validation is to randomly partition the training set S into two disjoint sets S', S_V . The subset S' is used to optimize the parameters θ_{fg} for a given model specification (f, g) . The complementary subset S_V , sometimes called the *validation set*, is used to estimate the risk of θ_{fg} :

$$\mathbf{E}_{\mathcal{D}}[\ell(h(x | \theta_{fg}), y)] \approx \frac{1}{|S_V|} \sum_{(x_i, y_i) \in S_V} \ell(h(x_i | \theta_{fg}), y_i). \quad (5.7)$$

This partitioning process is typically repeated several times and the results are averaged to reduce the variance of (5.7) introduced by sub-sampling the data. The validation procedure then selects θ_{fg} which achieves the lowest (average) validation error.

There are virtually countless variations on this validation procedure, such as cross-validation, stratified sampling, parameter grid search, and Bayesian hyper-parameter optimization [12, 13, 110]. A full survey of these techniques is beyond the scope of this chapter, but for our purposes, it is important to be comfortable with the concepts of validation, hyper-parameter optimization, and testing.

5.2 Discriminative Models

This section provides an overview of discriminative approaches to classification. Models will be described in terms of their objective functions, but we will omit the details of implementing specific optimization algorithms for parameter estimation.

In simple terms, a *discriminative model* seeks to predict a label y as a function of an input observation x , but does not explicitly model the input space \mathcal{X} . In this sense, discriminative models are simpler than *generative models* (Sect. 5.3), which must model the joint distribution over $\mathcal{X} \times \mathcal{Y}$. We will begin with an overview of binary linear models, extend them to multi-class models, and discuss their application to time-series data.

5.2.1 Binary Linear Models

The simplest models that practitioners regularly encounter are *linear models*. For binary classification problems with $\mathcal{X} \subseteq \mathbb{R}^d$, a linear model is parameterized by a weight vector $w \in \mathbb{R}^d$ and bias $b \in \mathbb{R}$, so that $\theta = (w, b)$. The model is linear in the

sense that the parameters w and b interact with the data through an inner product (and scalar addition) to produce a score $\langle w, x \rangle + b$. The output space is defined as $\mathcal{Y} = \{-1, +1\}$, so that the decision rule takes the form:

$$h(x | \theta) = \text{sign}(\langle w, x \rangle + b), \quad (5.8)$$

and the typical loss function of interest is the 0–1 loss. As mentioned in Sect. 5.1.1, the 0–1 loss is difficult to optimize directly, and different choices of surrogate functions lead to different models and algorithms.

5.2.1.1 Support Vector Machines

One of the simplest surrogate functions for the 0–1 loss is the *margin hinge loss*:

$$f_+(x, y | \theta) := \max(0, 1 - y(\langle w, x \rangle + b)), \quad (5.9)$$

which incurs 0 loss when the score $\langle w, x \rangle + b$ has the same sign as y —so that the prediction is correct—and its magnitude is at least 1 (the *margin*). The choice of 1 for the margin coincides with the error for misclassification $\ell(0, 1) = \ell(1, 0)$, and ensures that f_+ provides an upper bound on the 0–1 loss as illustrated in Fig. 5.1.

Combined with a quadratic penalty on w , the hinge loss gives rise to the standard linear *support vector machine* (SVM) [30]:

$$\min_{w, b} \frac{\lambda}{2} \|w\|^2 + \frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(\langle w, x_i \rangle + b)). \quad (5.10)$$

The hyper-parameter $\lambda > 0$ balances the trade-off between accuracy (minimizing loss) and model complexity (minimizing the norm of w).

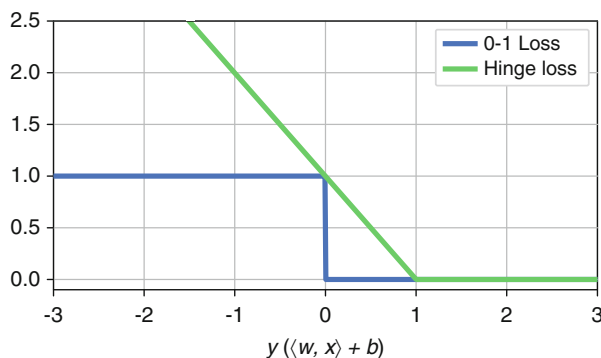


Fig. 5.1 The 0–1 loss and the hinge loss with a margin of 1. The hinge loss provides a continuous, convex upper bound on the 0–1 loss

5.2.1.2 Logistic Regression

An alternative to the SVM method in the previous section is to suppose a probabilistic model of the conditional probability $\mathbf{P}_\theta[y = +1 | x]$. Since the output space is binary, the Bernoulli distribution is a natural choice here:

$$\mathbf{P}[y = +1] := p \quad (5.11)$$

$$\mathbf{P}[y = -1] := 1 - p = 1 - \mathbf{P}[y = +1]$$

where $p \in [0, 1]$ is the probability of a positive label. To parameterize a Bernoulli distribution $\mathbf{P}_\theta[y = +1 | x]$ by the linear score function $\langle w, x \rangle + b$, the score can be mapped to the unit interval $[0, 1]$ via the *logistic function*:

$$\sigma(t) := \frac{1}{1 + e^{-t}}. \quad (5.12)$$

This results in the following conditional distribution for the label y given the input x :

$$\mathbf{P}_\theta[y = +1 | x] := \sigma(\langle w, x \rangle + b) = \frac{1}{1 + e^{-(w \cdot x) - b}} \quad (5.13)$$

$$\mathbf{P}_\theta[y = -1 | x] := 1 - \mathbf{P}_\theta[y = +1 | x].$$

As depicted in Fig. 5.2, the decision rule (5.8) coincides with choosing the most probable label under this model.

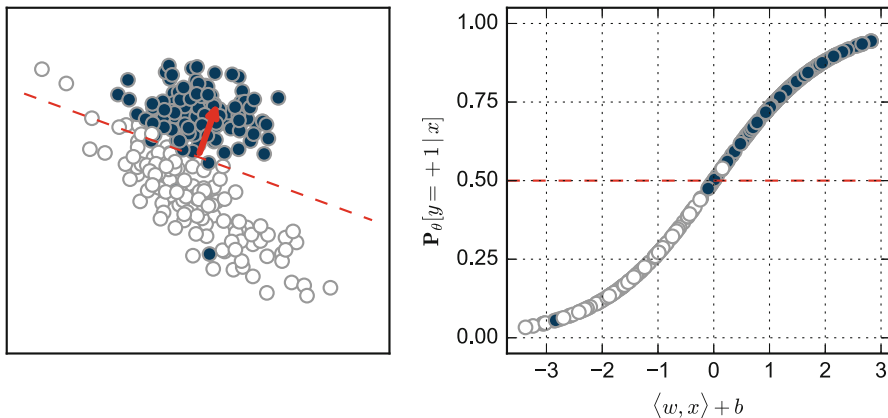


Fig. 5.2 An example of logistic regression in two dimensions. The *left plot* illustrates the data (*white and blue points*) and the learned linear model w (*red arrow*). The *right plot* illustrates the linear score $\langle w, x \rangle + b$ for each point x compared to the model probability $\mathbf{P}_\theta[y = +1 | x]$, where each point is *colored* according to its label. The decision threshold (0.5) is drawn in *red*

Taking the negative logarithm of (5.13) results in the following surrogate function:

$$\begin{aligned}
 f_{\sigma}(x_i, y_i | \theta) &:= \begin{cases} \log(1 + e^{-\langle w, x_i \rangle - b}) & y_i = +1 \\ \langle w, x_i \rangle + b + \log(1 + e^{-\langle w, x_i \rangle - b}) & y_i = -1 \end{cases} \\
 &= \left(\frac{1 - y_i}{2} \right) (\langle w, x_i \rangle + b) + \log(1 + e^{-\langle w, x_i \rangle - b}). \quad (5.14)
 \end{aligned}$$

Because of its use of the logistic function in defining (5.13), this formulation is known as *logistic regression* [32].

Although logistic regression and SVM share the same parameterization and have equivalent prediction rules, there are some key distinctions to keep in mind when deciding between the two methods. First, the scores produced by logistic regression have a natural probabilistic interpretation, whereas the SVM's scores do not directly correspond to a probabilistic model.³ Probabilistic interpretation can be useful when the classifier must produce confidence-rated predictions, or be integrated with larger models, such as the hidden Markov models discussed later in Sect. 5.3.5. Second, the choice of regularization $g(w)$ can have significant influence on the behavior of the model. While SVMs use the ℓ_2 (quadratic) penalty, logistic regression is often implemented with ℓ_1 or ℓ_2 penalties. The ℓ_2 penalty can be seen as limiting the influence of any single feature, while ℓ_1 can be seen as encouraging sparse solutions that depend only on a small number of features. In practice, the choice of regularization functions is another modeling decision that can be optimized for using cross-validation, since most common implementations of linear models support a range of penalty functions [44, 92].

5.2.2 Multi-Class Linear Models

The binary formulations in Sect. 5.2.1 can be naturally extended to the *multi-class* setting, where $\mathcal{Y} = \{1, 2, \dots, C\}$, so that each example is categorized into exactly one of the C distinct classes. Note that this is distinct from the similarly named *multi-label* setting, where each example can be assigned to multiple, non-disjoint classes. While the multi-label setting is often a natural fit for practical applications, it can be handled directly by using C independent binary classifiers.⁴

A natural extension of binary logistic regression can be obtained by defining $\theta = (w_c, b_c)_{c=1}^C$, so that each class has its own set of parameters (w_c, b_c) .

³SVM scores can be converted into probabilities via Platt scaling [94] or isotonic regression [122], but these methods require additional modeling and calibration.

⁴The notion of independence for multi-label problems will be treated more thoroughly when we develop deep learning models.

The probability that a given input x belongs to category j is then defined as

$$\mathbf{P}_\theta[y = j | x] := \frac{e^{\langle w_j, x \rangle + b_j}}{\sum_c e^{\langle w_c, x \rangle + b_c}}. \quad (5.15)$$

Taking the negative log-likelihood of the observed training data results in the following multi-class objective function:

$$f(x, y | \theta) := -\langle w_y, x \rangle - b_y + \log \left(\sum_c e^{\langle w_c, x \rangle + b_c} \right). \quad (5.16)$$

Similarly, the linear hinge loss can be generalized by comparing the discriminant score of the true label y for training point x to all other labels c [33]:

$$\begin{aligned} f(x, y | \theta) &:= \max \left(0, 1 - \langle w_y, x \rangle - b_y + \max_{c \neq y} \langle w_c, x \rangle + b_c \right) \\ &= -\langle w_y, x \rangle - b_y + \max_c \ell(y, c) + \langle w_c, x \rangle + b_c. \end{aligned} \quad (5.17)$$

Practically speaking, both objectives lead to the same prediction rule:

$$h(x | \theta) := \underset{y}{\operatorname{argmax}} \langle w_y, x \rangle + b_y, \quad (5.18)$$

that is, take the label with the highest score.

In multi-class problems, the regularization function is typically applied independently to each w_c and summed: $g(\theta) := \sum_c g_w(w_c)$.

5.2.3 Non-linear Discriminative Models

This section focused on linear models, primarily due to their simplicity, adaptability, and ease of integration with methods discussed in the remainder of this chapter. However, there are a wide range of effective, non-linear discriminative models available to practitioners, which we will briefly describe here. Interested readers are referred to [48] for thorough introductions to these methods.

Most closely related to the linear models described above are *kernel methods* [107]. These methods can be seen as implementing linear models after a non-linear transformation of the data encoded by a kernel function $k(x_1, x_2)$ which generalizes the notion of linear inner product $\langle x_1, x_2 \rangle$. Common choices of kernel functions include the *radial basis function* or *Gaussian kernel*:

$$k_\alpha(x_1, x_2) := \exp \{ -\alpha \|x_1 - x_2\|^2 \} \quad (5.19)$$

with bandwidth $\alpha > 0$, or the *polynomial kernel*:

$$k_{b,p}(x_1, x_2) := (b + \langle x_1, x_2 \rangle)^p \quad (5.20)$$

with degree $p \geq 1$ and constant $b \geq 0$. Kernel formulations are available for a broad class of suitably regularized models, including the SVM and ℓ_2 -regularized logistic regression [103].

Nearest neighbor classifiers [35, 47] operate by querying the training set \mathcal{X} for the nearest examples to a test example x , and predicting $h(x)$ as the majority vote of labels within the nearest neighbor set. This approach is simple to implement, and readily adapts to high-cardinality output label sets. The accuracy of nearest neighbor classifiers depends on the choice of distance function used to determine proximity, and there are a variety of methods available to optimize the metric from a labeled training set [9].

Finally, decision trees [22] operate by recursively partitioning the training set by applying a threshold to individual features. For example, the rule $x_3 \geq 0.75$ would send all examples with the third coordinate less than 0.75 to the left sub-tree, and all others to the right. Recursively applying these rules produces a tree structure, where each leaf of the tree is labeled according to the majority vote of training data that maps to that leaf. Test examples are then classified by the label of the leaf into which they map. Although decision trees are known to be prone to over-fitting, *random forests* [21] ameliorate this by combining the outputs of multiple trees to produce the final classifier. By generating an ensemble of trees from different (random) subsets of the training set and random subsets of features, a random forest tends to be much more robust than a single decision tree, and the general method is highly effective in practice.

5.3 Generative Models

The models in the previous section were discriminative, in the sense that they only need to describe the boundaries between categories, and not the distribution of data within each category. By contrast, *generative models* seek to approximate the data generating process itself by modeling the joint distribution $\mathbf{P}_\theta[x, y]$, rather than the conditional distribution $\mathbf{P}_\theta[y | x]$.

Before getting into specific examples of generative models, we will first cast the modeling process into the regularized optimization framework described at the beginning of this chapter, and provide a general overview of statistical inference and parameter estimation.

5.3.1 Maximum Likelihood Estimation

When building a generative model, the primary goal is to describe the space of observable data. Consequently, we should strive to make the model distribution \mathbf{P}_θ match the unknown data distribution \mathcal{D} , and our notion of *loss* is tied not to the accuracy of the resulting classifier, but to the *dissimilarity* between \mathbf{P}_θ and $\mathbf{P}_\mathcal{D}$. From information theory, a particularly useful notion of dissimilarity between probability distributions is the *Kullback–Leibler (KL) divergence* [31, 77]:

$$\text{KL}(\mathbf{P}_\mathcal{D} \parallel \mathbf{P}_\theta) := \int_z \log \left(\frac{\mathbf{P}_\mathcal{D}[z]}{\mathbf{P}_\theta[z]} \right) \mathbf{P}_\mathcal{D}[z] dz. \quad (5.21)$$

which measures the amount of information lost when using distribution \mathbf{P}_θ to approximate $\mathbf{P}_\mathcal{D}$: the more similar the two distributions are, the smaller the KL-divergence will be.

When \mathcal{D} is fixed, minimizing (5.21) over the choice of θ is equivalent to minimizing the cross-entropy between $\mathbf{P}_\mathcal{D}$ and \mathbf{P}_θ :

$$\operatorname{argmin}_\theta \text{KL}(\mathbf{P}_\mathcal{D} \parallel \mathbf{P}_\theta) = \operatorname{argmin}_\theta - \int_z \log(\mathbf{P}_\theta[z]) \mathbf{P}_\mathcal{D}[z] dz. \quad (5.22)$$

When \mathcal{D} is unknown, except through an I.I.D. sample $\{z_i\}_{i=1}^n \sim \mathcal{D}$, we can approximate (5.22) by the empirical average log-likelihood:

$$- \int_z \log(\mathbf{P}_\theta(z)) \mathbf{P}_\mathcal{D}[z] dz = -\mathbf{E}_\mathcal{D}[\log \mathbf{P}_\theta[z]] \approx -\frac{1}{n} \sum_{i=1}^n \log \mathbf{P}_\theta[z_i]. \quad (5.23)$$

This leads to the standard formulation of *maximum likelihood* parameter estimation: maximizing the probability of \mathbf{P}_θ generating the training data observed is approximately equivalent to minimizing the KL-divergence between \mathbf{P}_θ and $\mathbf{P}_\mathcal{D}$. For labeled observations $z = (x, y)$, the corresponding objective function f is then the negative log-likelihood given the model parameters θ :

$$f(x, y | \theta) = -\log \mathbf{P}_\theta[x, y]. \quad (5.24)$$

Once θ has been estimated, the prediction rule for an input example x then takes the form:

$$h(x | \theta) := \operatorname{argmax}_y \mathbf{P}_\theta[x, y]. \quad (5.25)$$

5.3.2 Bayesian Estimation: Maximum A Posteriori

In Sect. 5.3.1, there was no explicit mechanism to specify a preference for certain configurations of θ over others (aside from how well it approximates \mathcal{D}). The Bayesian approach to resolve this issue is to treat θ as a random variable, alongside

the observable data (x, y) . In this view, the model probability distribution \mathbf{P}_θ can be interpreted as conditional on a specific value of θ :

$$\mathbf{P}_\theta[x, y] := \mathbf{P}[x, y | \theta], \quad (5.26)$$

and we suppose a *prior distribution* $\mathbf{P}[\theta]$ to express preference for some values of θ over others. Similarly, the corresponding prediction rule for a given value of θ becomes

$$h(x | \theta) := \underset{y}{\operatorname{argmax}} \mathbf{P}[x, y | \theta]. \quad (5.27)$$

Bayesian inference consists of computing the *posterior distribution* $\mathbf{P}[\theta | S]$ after observing samples $S = \{(x_i, y_i)\}_{i=1}^n \sim \mathcal{D}$ by using Bayes' rule:

$$\mathbf{P}[\theta | S] = \frac{\mathbf{P}[S | \theta] \times \mathbf{P}[\theta]}{\mathbf{P}[S]}, \quad (5.28)$$

where $\mathbf{P}[S | \theta] = \prod_{i=1}^n \mathbf{P}[x_i, y_i | \theta]$ factorizes because S is assumed to be drawn I.I.D. Computing (5.28) is difficult because the denominator $\mathbf{P}[S]$ is an unknown quantity (i.e., \mathcal{D}) that is generally difficult to estimate. However, if we are only interested in finding a single value of θ which maximizes (5.28), then the $\mathbf{P}[S]$ factor may be safely ignored, since it is constant with respect to the choice of θ . This leads to the *maximum a posteriori* (MAP) formulation of parameter estimation:

$$\underset{\theta}{\operatorname{argmin}} -\frac{1}{n} \sum_{i=1}^n \log \mathbf{P}[x_i, y_i | \theta] - \frac{1}{n} \log \mathbf{P}[\theta]. \quad (5.29)$$

This is derived by taking the logarithm of (5.28), which is equivalent to maximum likelihood inference (5.23), but with an additive term $g(\theta) := -\frac{1}{n} \log \mathbf{P}[\theta]$.⁵ MAP inference can thus be viewed as a special case of the generic regularized learning objective (5.6).

The choice of prior distribution $\mathbf{P}[\theta]$ is of utmost importance, and generally depends on several contributing factors such as model structure, existing domain knowledge, and computational convenience. In the following sections, we will discuss the choice of priors for specific models.

5.3.3 *Aside: Fully Bayesian Inference*

The MAP estimation approach described in the previous section results in classifiers that depend on a single value of θ . If the posterior distribution $\mathbf{P}[\theta | S]$ is not strongly

⁵The factor of $1/n$ is not strictly necessary here, but are included for consistency with (5.6).

peaked, or has multiple modes which result in disagreeing predictors, then MAP estimation can become unstable. These situations call for *fully Bayesian inference*, where the uncertainty in θ is explicitly accounted for when making predictions.

Instead of (5.27), a fully Bayesian predictor would marginalize θ out of the joint distribution $\mathbf{P}[x, y, \theta]$ to find the most likely label:

$$h(x) := \underset{y}{\operatorname{argmax}} \mathbf{P}[x, y] = \underset{y}{\operatorname{argmax}} \int_{\theta} \mathbf{P}[x, y | \theta] \times \mathbf{P}[\theta] d\theta. \quad (5.30)$$

In general, this marginal likelihood calculation does not have a closed-form solution, and it can therefore be difficult to compute exactly. When fully Bayesian inference is necessary, it is typically performed by sampling methods such as Markov chain Monte Carlo (MCMC) [62, 87], which can estimate $\mathbf{P}[x, y]$ by drawing samples $\theta \sim \mathbf{P}[\theta]$ and averaging the likelihood estimates $\mathbf{P}[x, y | \theta]$. Once the posterior distribution (5.28) has been computed from a training set S , (5.30) can be approximated by sampling from the posterior $\mathbf{P}[\theta | S]$ rather than the prior $\mathbf{P}[\theta]$.

There is a rich literature on sampling methods for marginal likelihood, and these methods lie outside the scope of this text [4, 50]. For the remainder of this chapter, we will stick primarily with MAP inference for probabilistic models.

5.3.4 Gaussian Mixture Models

A *Gaussian mixture model* (GMM) consists of a weighted mixture of K multivariate Gaussian distributions [91]. Formally, $\theta = (\omega_k, \mu_k, \Sigma_k)_{k=1}^K$ where ω_k are non-negative weights which sum to 1, and $\mu_k \in \mathbb{R}^d$ and $\Sigma_k \in \mathbb{S}_{++}^d$ denote the mean vector and covariance matrix of the k th mixture component.⁶ The probability density at point x is then defined as:

$$\begin{aligned} \mathbf{P}_{\theta}[x] &:= \sum_k \omega_k \times \mathcal{N}(\mu_k, \Sigma_k) \\ &= \sum_k \omega_k \times |2\pi \Sigma_k|^{-1/2} \times e^{-\frac{1}{2} \|x - \mu_k\|_{\Sigma_k}^2}, \end{aligned} \quad (5.31)$$

where $\|z\|_{\Sigma}^2 := z^T \Sigma^{-1} z$. Given a sample $(x_i)_{i=1}^n$, the parameters θ can be inferred by a variety of different strategies, but the most common method is expectation-maximization (EM) [36].

⁶ \mathbb{S}_{++}^d denotes the set of $d \times d$ positive definite matrices: Hermitian matrices with strictly positive eigenvalues.

5.3.4.1 Classification with GMMs

Note that (5.31) does not involve the labels y , and can therefore be considered an *unsupervised* model of the data. This can be extended to a multi-class supervised model by fitting a separate GMM $P_{\theta_y}[x|y]$ for each category y . The objective then becomes to find GMM parameters $\theta = (\theta_y, p_y)$, where θ_y contains the parameters of the GMM corresponding to class y , and p_y models the probability of class y . Given an unlabeled example x , the label is predicted as

$$h(x | \theta) := \operatorname{argmax}_y \mathbf{P}_\theta[y | x] = \operatorname{argmax}_y \mathbf{P}_\theta[x | y] \times \mathbf{P}_\theta[y], \quad (5.32)$$

where the latter equality follows from Bayes' rule:

$$\mathbf{P}_\theta[y | x] = \frac{\mathbf{P}_\theta[x | y] \times \mathbf{P}_\theta[y]}{\mathbf{P}_\theta[x]} \propto \mathbf{P}_\theta[x | y] \times \mathbf{P}_\theta[y] \quad (5.33)$$

because $\mathbf{P}_\theta[x]$ is (an unknown) constant when searching over y for a given x . The interpretation of (5.32) is similar to that of the multi-class linear models of the previous section: the predicted label is that for which the corresponding generative model assigns highest probability to the input x .

5.3.4.2 Simplifications

There are a few commonly used simplifications to the GMM described in (5.31), as illustrated in Fig. 5.3. The first simplification is to restrict the parameter space so that each Σ_k is a diagonal matrix. This reduces the number of parameters in the model, and simplifies the matrix inverse and determinant calculations in (5.31). This restriction prohibits the model from capturing correlations between variables. However, if the training data has already been decorrelated by a pre-processing step (such as principal components analysis), the diagonal restriction may perform well in practice.

Spherical covariance constraints force $\Sigma_k = \sigma_k I_k$, so that each component has equal variance along each dimension, but that variance can differ from one component to the next. An even more extreme constraint is to force all Σ_k to equal the identity matrix. This restriction, known as *isotropic covariance*, eliminates all variance parameters from the model, so all that are left are the mixture coefficients ω_k and the means μ_k . The spherical restriction may be justified if in addition to being decorrelated, the data are pre-processed to have unit variance along each coordinate, and variance is expected to be independent of component membership. In this case, the GMM can be interpreted as a soft-assignment variant of the K-means clustering algorithm [82].

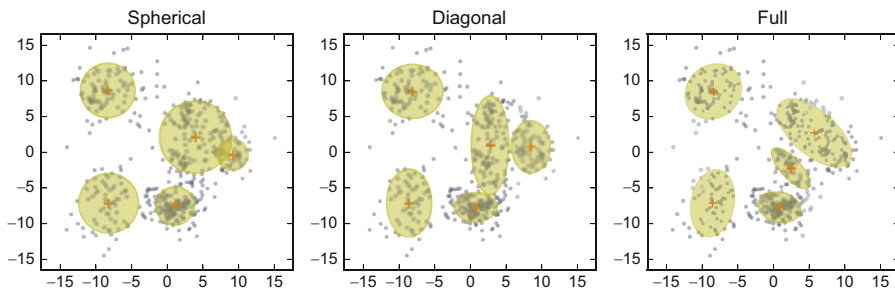


Fig. 5.3 Gaussian mixture models with different covariance constraints applied to the same data set (blue). Component means μ_k are indicated by red plus markers, and the covariance structures are indicated by yellow ellipses covering ± 3 standard deviations from μ_k

5.3.4.3 Aside: Maximum Likelihood, or MAP?

As described in the introduction to this section, we have a choice between classical (maximum likelihood) and Bayesian (MAP) inference when estimating the parameters θ of a generative model. For the GMM as described above in (5.31), it should be noted that the classical approach has certain degeneracies that can be avoided by the Bayesian approach.

Specifically, given a training sample, it is possible to make the likelihood arbitrarily large by selecting one component (μ_k, Σ_k) and setting $\mu_k = x_i$ (for some training point x_i), and letting $\Sigma_k = \lambda I$ for some arbitrarily small value $\lambda > 0$ so that the determinant $|\Sigma_k|$ approaches 0. Although it may be rare to encounter this degenerate case in practice, it is possible—especially when the training sample contains outliers (examples far in feature space from most of the training samples). Similar degeneracies can occur when modes of the data lie close to a low-rank subspace. This suggests that maximum likelihood inference may not be the most appropriate choice for estimating GMM parameters.

This situation can be avoided by incorporating prior distributions on the model parameters ω, μ_k, Σ_k which assign low probability to known degenerate configurations. The choice of prior distributions should be guided by domain knowledge and some conceptual understanding of the data, so any general-purpose recipes should be taken as suggestions and treated with skepticism. That said, for computational reasons, it is often preferable to use *conjugate priors*, which can lead to simple parameter updates and computationally efficient learning algorithms.⁷

⁷A probability distribution $\mathbf{P}[\theta]$ is a conjugate prior if the posterior $\mathbf{P}[\theta | S]$ has the same form as the prior $\mathbf{P}[\theta]$ [97].

In the case of the GMM, there are three prior distributions in need of definition: $\mathbf{P}[\omega]$, $\mathbf{P}[\Sigma]$, and $\mathbf{P}[\mu]$. Because ω is a categorical random variable, the (symmetric) Dirichlet distribution can be used since it is conjugate to the categorical distribution:

$$\mathbf{P}_\alpha[\omega] := \frac{\Gamma(\alpha K)}{\Gamma(\alpha)^K} \prod_{k=1}^K \omega_k^{\alpha-1}. \quad (5.34)$$

The $\alpha > 0$ variable is a hyper-parameter: for $\alpha \geq 1$, ω tends to be dense; for $\alpha < 1$, ω tends to concentrate on a few components, which can be used to effectively eliminate unused components.

The covariance prior $\mathbf{P}[\Sigma]$ can be somewhat more difficult to define. For diagonal covariance models, it is common to factor the prior over each variance component $\mathbf{P}[\Sigma] = \mathbf{P}[\sigma_i^2]$, and use a prior with support limited to positive reals, such as the log-normal or gamma distributions. If the prior assigns 0 probability to $\sigma_i^2 = 0$ —as the log-normal distribution does, or gamma with shape parameter $\alpha > 1$ —then the degenerate cluster issue described above can be effectively prevented. For full-covariance models, the Wishart distribution (a multivariate extension of the gamma distribution with support over positive definite matrices) can be used to achieve similar results. Each of these prior distributions has additional hyper-parameters which specify the location and dispersion of probability mass.

Finally, the prior on cluster means $\mathbf{P}[\mu]$ is often taken to be a standard, multivariate Gaussian when $\mathcal{X} = \mathbb{R}^d$. However, if additional constraints are known, e.g., observations are non-negative magnitude spectra so $\mathcal{X} = \mathbb{R}_+^d$, then a coordinate-wise log-normal or gamma distribution might be more appropriate. For a more thorough discussion of priors for generative models, we refer interested readers to [85, Chap. 5].

5.3.4.4 Parameter Estimation

While it is relatively straightforward to implement the expectation-maximization (EM) algorithm for the maximum likelihood formulation of a GMM, the task can become significantly more complex in the MAP scenario, as the update equations may no longer have convenient, closed-form solutions. Variational approximate inference methods are often used to simplify the parameter estimation problem in this kind of setting, usually by computing the MAP solution under a surrogate distribution with a more computationally convenient factorization [118]. A proper treatment of variational inference is beyond the scope of this text, and it should be noted that although the resulting algorithms are “simpler” (more efficient), the derivations can be more complex as well. Software packages such as Stan [23] and Edward [114] can ease the burden of implementing variational inference by automating much of the tedious calculations.

5.3.4.5 How Many Components?

Throughout this section, we have assumed that the number of mixture components K was fixed to some known value. In practice, however, the best K is never known a priori, so practitioners must find some way to select K . There are essentially three data-driven approaches to selecting K : information criteria, Dirichlet process mixtures, and utility optimization.

The first approach comes in a wide range of flavors: Akaike’s information criterion (AIC) [2], Bayesian information criterion (BIC) [105], or widely applicable information criterion (WAIC) [119]. The common thread throughout these methods is that one first constructs a set of models—for a GMM, each model would correspond to a different choice of K —and select the one which best balances accuracy (likelihood of observed data) against model complexity. The methods differ in how “model complexity” is estimated, and we refer interested readers to Watanabe [119] for a survey of the topic.

The second approach, Dirichlet process mixtures [5], implicitly supports a countably infinite number of mixture components, and estimates K as another model parameter along with mixing weights, means, and variances [15, 98]. This model can be approximated by setting K to some reasonable upper limit on the acceptable number of components, imposing a sparse Dirichlet prior ($\alpha < 1$) over ω , and estimating the parameters just as in the case where K is fixed [69]. Then, any components with sufficiently small mixture weights ω_i (e.g., those whose combined weight is less than 0.01) can be discarded with negligible impact on the corresponding mixture density.

Finally, utility-based approaches select the model which works best for a given application, i.e., maximizes some expected utility function. In classification problems, the natural utility function to use would be classification accuracy of the resulting predictor (5.32). Concretely, this would amount to treating K as another hyper-parameter to be optimized using cross-validation, with classification accuracy as the selection criterion.

5.3.5 Hidden Markov Models

So far in this chapter, the models described have not directly addressed temporal dynamics of sound. To model dynamics, we will need to treat a sequence of feature observations as a single object, which we will denote by $x = (x[1], x[2], \dots, x[T])$.⁸

⁸Note that although we use T to denote the length of an arbitrary sequence x , it is not required that all sequences have the same length.

In general, the likelihood of a sequence observation $\mathbf{P}_\theta[x]$ can be factored as follows:

$$\begin{aligned}\mathbf{P}_\theta[x] &= \mathbf{P}_\theta[x[1], x[2], \dots, x[T]] \\ &= \mathbf{P}_\theta[x[1]] \times \prod_{t=1}^{T-1} \mathbf{P}_\theta[x[t+1] | x[1], x[2], \dots, x[t]].\end{aligned}\quad (5.35)$$

The *Markov* assumption asserts that this distribution factors further:

$$\mathbf{P}_\theta[x] := \mathbf{P}_\theta[x[1]] \times \prod_{t=1}^{T-1} \mathbf{P}_\theta[x[t+1] | x[t]].\quad (5.36)$$

That is, the distribution at time $t+1$ conditional on time t is independent of any previous time $t' < t$. A *hidden Markov model* (HMM) asserts that all dynamics are governed by hidden discrete “state” variables $z[t] \in \{1, 2, \dots, K\}$, and that an observation $x[t]$ at time t depends only on the hidden state $z[t]$ [7, 96].

Formally, an HMM is defined by a joint distribution of the form:

$$\mathbf{P}_\theta[x, z] := \prod_{t=1}^T \mathbf{P}_\theta[z[t] | z[t-1]] \times \mathbf{P}_\theta[x[t] | z[t]].\quad (5.37)$$

There are three distinct components to (5.37):

- $\mathbf{P}_\theta[z[1] | z[0]]$ is the *initial state model*, which determines the probability of starting a sequence in each state⁹;
- $\mathbf{P}_\theta[z[t] | z[t-1]]$ is the *transition model*, which governs how one hidden state transitions to the next; and
- $\mathbf{P}_\theta[x[t] | z[t]]$ is the *emission model*, which governs how each hidden state generates observed data.

If there are K possible values for a hidden state, then the transition model can be defined as a collection of K categorical distributions over the K hidden states. The parameters of these distributions are often collected into a $K \times K$ stochastic matrix known as the *transition matrix* V , where

$$V_{ij} = \mathbf{P}_\theta[z[t] = i | z[t-1] = j].\quad (5.38)$$

Similarly, the initial state model can also be defined as a categorical distribution over the K hidden states.

⁹For ease of notation, we denote the initial state distribution as $\mathbf{P}_\theta[z[1] | z[0]]$, rather than the unconditional form $\mathbf{P}_\theta[z[1]]$.

The definition of the emission model depends on the form of the observed data. A common choice, when $x[t] \in \mathbb{R}^d$ is to define a multivariate Gaussian emission model for each hidden state:

$$\mathbf{P}_\theta [x | z[t] = k] := \mathcal{N}(\mu_k, \Sigma_k). \quad (5.39)$$

This model is commonly referred to as the Gaussian-HMM. Note that the specification of the emission model does not depend on the transition model, and any reasonable emission model may be used instead. Emission models can themselves also be mixture models, and GMMs are particularly common.

Once the parameters of the model have been estimated (Sect. 5.3.5.2), the most likely hidden state sequence z for an observed sequence x can be inferred by the Viterbi algorithm [117]. The resulting state sequence can be used to segment the sequence into contiguous subsequences drawn from the same state. In audio applications, this can correspond directly to the temporal activity of a class or sound source [64].

5.3.5.1 Discriminative HMMs

The most common way to apply HMMs for classification is to impose some known structure over the hidden state space. For example, in speech recognition applications, we may prefer a model where each hidden state corresponds to a known phoneme [73]. If labeled training data is available, where each observation sequence $x = (x[1], x[2], \dots, x[T])$ has a corresponding label sequence $y = (y[1], y[2], \dots, y[T])$, then we can directly relate the hidden state space to the label space \mathcal{Y} . While one could use a discriminative model to independently map each observation to a label, this would ignore the temporal dynamics of the problem. Integrating the classification with an HMM can be seen as a way of imposing temporal dynamics over model predictions.

Recall that there are three quantities to be estimated in an HMM: the initial state distribution, the state transition distribution, and the emission distribution. When labeled training data is available—i.e., the state variable for each observation is also observed—the first two distributions can be estimated directly from the labels, since they are conditionally independent of the input data x given the state. In practice, this amounts to estimating the parameters of $K + 1$ categorical distributions— K for the transition distributions and one for the initial state distribution—from the observed labeled sequences.

All that remains is to characterize the emission distributions. This can be done by applying Bayes' rule to the observation model, now using y to indicate states instead of z :

$$\mathbf{P}_\theta [x[t] | y[t] = k] = \frac{\mathbf{P}_\theta [y[t] = k | x[t]] \times \mathbf{P}_\theta [x[t]]}{\mathbf{P}_\theta [y[t] = k]} \quad (5.40)$$

which expresses the emission probability in terms of the conditional class likelihood $\mathbf{P}_\theta [y[t] = k | x[t]]$, the marginal probability of the class occurring $\mathbf{P}_\theta [y[t] = k]$, and the marginal probability of the observation $\mathbf{P}_\theta [x[t]]$. The conditional class likelihood can be estimated by any probabilistic discriminative classifier, e.g., logistic regression (Sect. 5.2.1.2) or a multi-layer perceptron (Sect. 5.4.2). The marginal probability of the class is a categorical distribution that can be estimated according to the statistics of the labeled training data.

Finally, the marginal probability of the observation $\mathbf{P}_\theta [x[t]]$ is generally difficult to estimate, but luckily it is not often needed. Recall that the practical application of the HMM is to produce a sequence of labels $y = (y[1], y[2], \dots, y[T])$ from an unlabeled observation sequence $x = (x[1], x[2], \dots, x[T])$ following the prediction rule:

$$h(x | \theta) := \underset{y}{\operatorname{argmax}} \mathbf{P}_\theta [x, y]. \quad (5.41)$$

Substituting (5.40) and (5.37) into (5.41) yields

$$\mathbf{P}_\theta [x, y] = \prod_{t=1}^T \mathbf{P}_\theta [y[t] | y[t-1]] \times \mathbf{P}_\theta [x[t] | y[t]] \quad (5.42a)$$

$$= \prod_{t=1}^T \mathbf{P}_\theta [y[t] | y[t-1]] \times \mathbf{P}_\theta [x[t]] \times \frac{\mathbf{P}_\theta [y[t] | x[t]]}{\mathbf{P}_\theta [y[t]]} \quad (5.42b)$$

$$= \left(\prod_{t=1}^T \mathbf{P}_\theta [y[t] | y[t-1]] \times \frac{\mathbf{P}_\theta [y[t] | x[t]]}{\mathbf{P}_\theta [y[t]]} \right) \times \prod_{t=1}^T \mathbf{P}_\theta [x[t]] \quad (5.42c)$$

$$\propto \prod_{t=1}^T \mathbf{P}_\theta [y[t] | y[t-1]] \times \frac{\mathbf{P}_\theta [y[t] | x[t]]}{\mathbf{P}_\theta [y[t]]}, \quad (5.42d)$$

where the $\mathbf{P}_\theta [x[t]]$ factors can be ignored since they do not affect the maximization over choice of y . Consequently, the sequence prediction (5.41) can be computed by the Viterbi algorithm using only the discriminative classifier's point-wise output and the empirical unigram- and bigram-statistics, $\mathbf{P}_\theta [y[t]]$ and $\mathbf{P}_\theta [y[t] | y[t-1]]$, of observed label sequences.

In addition to attaching specific meaning to the “hidden” state variables (i.e., correspondence with class labels), there are two computational benefits to this approach. First, it can be applied to any probabilistic classifier, and it is often used as a post-processing technique to reduce errors resulting from frame-wise classifiers. Second, discriminative classifiers are often easier to train than generative models, since they typically require less observation data, and the resulting models tend to be more accurate in practice.

The discriminative HMM approach described here can be viewed as a special case of a conditional random field (CRF) [78], where the model parameters have been estimated independently. A more general CRF-based approach would jointly estimate all model parameters, which can improve accuracy in practice.

5.3.5.2 Priors and Parameter Estimation

Parameter estimation for HMMs can be done either in maximum likelihood or MAP formulations, and the resulting algorithms are qualitatively similar to the case for Gaussian mixture models.¹⁰ In particular, a Bayesian formulation of the HMM looks nearly identical to that of the GMM—with the initial state model $\mathbf{P}[z[1] | z[0], \theta]$ acting in place of the mixture weights $\mathbf{P}[\omega | \theta]$ —and the only additional set of parameters in need of a prior is the transition matrix. Since each row $V_{\cdot j}$ of the transition matrix is a categorical distribution, it is again natural to impose a Dirichlet prior over each row of V . For details on Bayesian HMM inference, we refer readers to Beal [8].

Just as in the GMM case, the number of hidden states K is another hyperparameter to be estimated, and it can be done via any of the methods described in Sect. 5.3.4.5. In the discriminative case where hidden states are matched to observable labels, this issue does not arise.

5.4 Deep Models

In this section, we provide a brief overview of the so-called *deep learning* architectures. While the term *deep learning* can apply to a wide range of different types of models, we will focus specifically on discriminative classification models which include a non-linear transformation of input data that is jointly optimized with the classifier. For a more thorough introduction, we refer interested readers to Goodfellow et al. [54].

5.4.1 Notation

Deep models are often characterized by compositions of non-linear functions. For brevity, we will denote the sequential composition of k functions by the \odot symbol, defined as:

$$\left(\odot_{i=1}^m f_i\right)(x) := (f_m \circ f_{m-1} \circ \cdots \circ f_1)(x). \quad (5.43)$$

Each f_i should be interpreted as a stage or layer of processing, which transforms the output of f_{i-1} to the input of f_{i+1} .

¹⁰The well-known Baum–Welch algorithm for HMM parameter estimation is a special case of expectation-maximization [96].

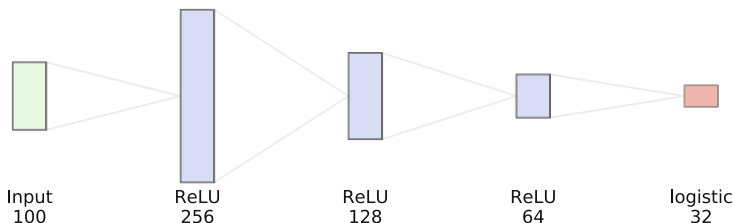


Fig. 5.4 An example illustration of a multi-layer perceptron (MLP) with four layers. The input $x \in \mathbb{R}^{100}$ is mapped through three intermediate layers with $d_1 = 256$, $d_2 = 128$, and $d_3 = 64$ and rectified linear unit (ReLU) transfer functions. The output layer in this example maps to 32 independent classes with a logistic transfer function

5.4.2 Multi-Layer Perceptrons

The simplest, and oldest family of “deep” models is the multi-layer perceptron (MLP) [99, 101]. As illustrated in Fig. 5.4, an MLP is defined by a sequence of *layers*, each of which is an affine transformation followed by a non-linear *transfer function* ρ :

$$f_i(z | \theta) := \rho_i(w_i^\top z + b_i), \quad (5.44)$$

where the parameters $\theta = (w_i, b_i, \rho_i)_{i=1}^m$ have weights $w_i \in \mathbb{R}^{d_{i-1} \times d_i}$, biases $b_i \in \mathbb{R}^{d_i}$, and transfer functions $\rho_i: \mathbb{R}^{d_i} \rightarrow \mathbb{R}^{d_i}$. Each layer maps data from $\mathbb{R}^{d_{i-1}}$ to \mathbb{R}^{d_i} , which dictates the shape of the resulting MLP.¹¹ For input data $x \in \mathbb{R}^d$, we define $d_0 := d$, and for categorical prediction tasks, we define $d_m := |\mathcal{Y}|$ as the number of labels.

The final (output) layer f_m of an MLP is typically defined as a linear model in one of the forms described in Sect. 5.2. The “internal” layers $f_1 \dots f_{m-1}$ can be interpreted as a “feature extractor.” One motivation for this architecture is that by jointly optimizing the internal and output layers, the model finds a feature extractor which separates the data so that the output layer performs well. In contrast to the linear models described in Sect. 5.2, this approach directly benefits from multi-class and multi-label data because it can leverage observations from all classes in constructing the shared representation.

The surrogate loss f is defined in terms of the output of the final layer:

$$f(x, y | \theta) := f_{\text{err}} \left(\left(\bigcirc_{i=1}^m f_i \right) (x), y \right), \quad (5.45)$$

¹¹Some authors refer to the layer dimension d_i as *width*. This terminology can be confusing when applied to spatio-temporal data as in Sect. 5.4.3, so we will use *dimension* to indicate d_i and retain *width* to describe a spatial or temporal extent of data.

where f_{err} is a standard surrogate loss function as described in Sect. 5.2 that compares the output of the final layer f_m to the target label y . Just as in the previous sections, the prediction rule corresponding to (5.45) is to choose the label which would minimize the objective:

$$h(x | \theta) := \underset{y}{\operatorname{argmin}} f(x, y | \theta). \tag{5.46}$$

Typically, this will simplify to an argmax over the output variables (for multi-class problems), or a thresholding operation (for binary problems).

5.4.2.1 Transfer and Objective Functions

The transfer function ρ_i —also known as an *activation function* or *non-linearity*—allows the model to learn non-linear structure in data.¹² As illustrated in Fig. 5.5, there are a variety of commonly used transfer functions.

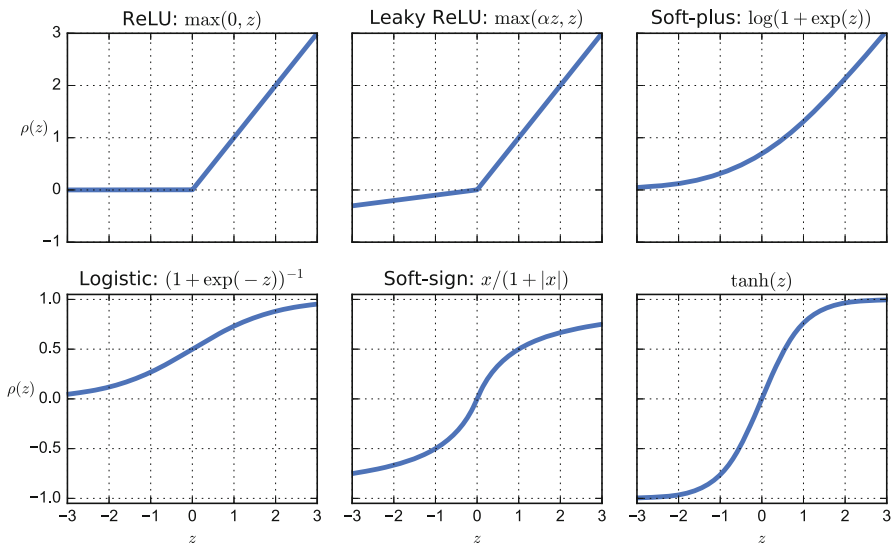


Fig. 5.5 A comparison of various transfer functions ρ . Note that some saturate on both negative and positive inputs (logistic, tanh, soft-sign), while others saturate only on negative inputs (ReLU, soft-plus), or not at all (leaky ReLU)

¹²To see this, observe that if ρ_i is omitted, then the full model $f(x | \theta)$ is a composition of affine functions, which is itself an affine function, albeit one with rank constraints imposed by the sequence of layer dimensions.

The choice of ρ_i for internal layers ($i < m$) is generally at the practitioner's discretion. As illustrated in Fig. 5.5, some non-linearities approach saturating values, such as tanh going to ± 1 , or logistic going to $\{0, 1\}$ as z diverges from 0. When the non-linearity saturates to a constant, its derivative goes to 0 in that region of its domain, and as a result, the error signal cannot (easily) propagate through to earlier layers. Although techniques exist to limit this behavior (e.g., by scaling the activations to stay within the non-saturating regions [68]), it is simpler in practice to use a one-sided or non-saturating transfer function, such as the *rectified linear unit* (ReLU) [86] or leaky ReLU [81].

Typically the choice of ρ_m (the output layer) is dictated by the structure of the output space. If the output is a multi-label prediction, then the logistic function ($\rho_m = \sigma$) provides a suitable transfer function that can be interpreted as the likelihood of each label given the input data. In this setting, the label is usually encoded as a binary vector $y \in \{0, 1\}^C$ (for C labels), and the standard loss function is the sum of log-likelihoods for each label:

$$f_{\text{err}}(\hat{y}, y) := \sum_{c=1}^C -y_c \log \hat{y}_c - (1 - y_c) \log (1 - \hat{y}_c), \quad (5.47)$$

where $\hat{y} = (\odot_{i=1}^m f_i)(x)$ is the output of the MLP on input x . This loss function is also known as the *binary cross-entropy* loss, since it is equivalent to the sum of cross-entropies between K pairs of Bernoulli random variables.

For multi-class problems, the soft-max function provides a normalized, non-negative output vector that can be interpreted as a categorical probability distribution:

$$\rho_{\text{softmax}}(z)_k := \frac{\exp(z_k)}{\sum_j \exp(z_j)}. \quad (5.48)$$

In multi-class problems, the label y is typically encoded as a binary vector with exactly one non-zero entry. The standard loss function is the categorical cross-entropy:

$$f_{\text{err}}(\hat{y}, y) := - \sum_{c=1}^C y_c \log \hat{y}_c. \quad (5.49)$$

5.4.2.2 Initialization

Related to the choice of transfer function is the issue of weight initialization. Because gradients do not propagate when the input to the transfer function lies in its saturating regions, it is beneficial to randomly initialize weights w_i and biases b_i such that $\mathbf{E}_{w_i, b} [\rho(w_i^\top z + b_i)]$ has non-zero derivative. Glorot and Bengio [52] derive an initialization scheme using weights w_{ij} sampled randomly from the

interval $\pm d_{i-1}^{-1/2}$, with the implicit assumption that the input z is bounded in $[-1, 1]$, as is the case when using symmetric, saturating transfer functions (e.g., logistic or tanh).

He et al. [63] argue that this scheme is ill-suited for networks in which the input z has non-zero expectation, as is the case for networks with ReLU activations. Instead, He et al. recommend that weights be initialized as $w_{ij} \sim \mathcal{N}\left(0, \sqrt{2/d_{i-1}}\right)$ for ReLU networks, or more generally,

$$w_{ij} \sim \mathcal{N}\left(0, \sqrt{\frac{2}{(1 + \alpha^2)d_{i-1}}}\right) \quad (5.50)$$

for leaky ReLU networks with parameter $\alpha \geq 0$.

We note that most common implementations provide these initialization schemes by default [1, 27, 38], but it is still up to the practitioner to decide which initialization to use in conjunction with the choice of transfer function.

5.4.2.3 Learning and Optimization

In general, the form of f does not lend itself to closed-form solutions, so the parameters are estimated by iterative methods, usually some variation of gradient descent:

$$\theta \mapsto \theta - \eta \nabla_{\theta} f(x, y | \theta) \quad (5.51)$$

where ∇_{θ} denotes the gradient operator with respect to parameters θ , and $\eta > 0$ is a learning rate that controls how far to move θ from one iteration to the next.

Because f is defined as a composition of simpler functions, the gradient $\nabla_{\theta} f$ is decomposed into its individual components (e.g., ∇_{w_i} or ∇_{b_j}), which are computed via the chain rule. This process is also known as *back-propagation*, since the calculation can be seen as sending an error signal back from the output layer through the sequence of layers in reverse-order [101]. In the past, calculating the gradients of (5.45) was a tedious, mechanical chore that needed to be performed for each model. However, in recent years, nearly all deep learning frameworks include automatic differentiation tools, which remove this burden of implementation [1, 11, 28, 71].

Due to the computational and memory complexity of computing gradients over a large training set, the common practice is to use *stochastic gradient descent* (SGD) [18], which estimates the gradient direction at each step k by drawing a small *mini-batch* B_k of samples from the training set S , and approximating the gradient:

$$\hat{\nabla}_{\theta} f := \frac{1}{|B_k|} \sum_{(x,y) \in B_k} \nabla_{\theta} f(x, y | \theta) \quad (5.52a)$$

$$\approx \frac{1}{|S|} \sum_{(x,y) \in S} \nabla_{\theta} f(x, y | \theta) = \nabla_{\theta} f \quad (5.52b)$$

$$\approx \mathbf{E}_{\mathcal{D}} [\nabla_{\theta} f(x, y | \theta)]. \quad (5.52c)$$

It is also common to accelerate the standard SGD approach described above by using *momentum* methods [88, 95, 111], which re-use gradient information from previous iterations to accelerate convergence. Similarly, adaptive update schemes like AdaGrad [41] and ADAM [75] reduce the dependence on the step size η , and can dramatically improve the rate of convergence in practice.

Finally, to prevent over-fitting of MLP-based models, it is common to use *early stopping* as a form of regularization [109], rather than minimizing (5.45) over the training set until convergence. This is usually done by periodically saving check-points of the model parameters θ , and then validating each check-point on held-out data as described in Sect. 5.1.2.

5.4.2.4 Discussion: MLP for Audio

Multi-layer perceptrons form the foundation of deep learning architectures, and can be effective across a wide range of domains. However, the MLP presents some specific challenges when used to model audio.

First, and this is common to nearly all models discussed in this chapter, is the choice of input representation. Practitioners generally have a wide array of input representations to choose from—time-domain waveforms, linear-frequency spectrograms, log-frequency spectrograms, etc.—and this choice influences the efficacy of the resulting model. Moreover, the scale of the data matters, as described in Sect. 5.4.2.2. This goes beyond the simple choice of linear or logarithmic amplitude spectrogram scaling: as discussed in the previous section, training is difficult when transfer functions operate in their saturating regions. A good heuristic is to scale the input data such that the first layer’s transfer function stays within non-saturating region in expectation over the data. Coordinate-wise standardization (also known as *z-scoring*) using the training set’s mean and variance statistics (μ, σ^2) accomplishes this for most choices of transfer functions, since each coordinate maps most of the data to the range $[-3, +3]$ ¹³:

$$x_i \mapsto \frac{x_i - \mu_i}{\sigma_i}. \quad (5.53)$$

In practice, coordinate-wise standardization after a log-amplitude scaling of spectral magnitudes works well for many audio applications.

¹³Note that batch normalization accomplishes this scaling implicitly by estimating these statistics during training [68].

Second, an MLP architecture requires that all input have a fixed dimension d , which implies that all audio signals must be cropped or padded to a specified duration. This is typically achieved by dividing a long signal (or spectrogram) $x \in \mathbb{R}^{T \times n}$ into small, fixed-duration observations $x_i \in \mathbb{R}^{\delta \times n}$. Observations x_i can be interpreted as vectors of dimension $d = n\delta$, and processed independently by the MLP. In doing so, care must be taken to ensure that the window length δ is sufficiently long to capture the target concept.

Finally, MLPs do not fully exploit the structure of audio data implicit in time or frequency dimensions. For example, if two observations x_1, x_2 are derived from a signal spanning frame indices $[t, t + \delta]$ and $[t + 1, t + \delta + 1]$, respectively, the MLP outputs for $f(x_1)$ and $f(x_2)$ can diverge significantly, even though the inputs differ only by two frames. Consequently, MLPs trained on audio data can be sensitive to the relative positioning of an observation within the window. For non-stationary target concepts, this presents a great difficulty for MLP architectures, since they effectively need to detect the target event at every possible alignment within the window. The remaining sections of this chapter describe methods to circumvent this problem by exploiting the ordering of time or frequency dimensions.

5.4.3 Convolutional Networks

Convolutional networks are explicitly designed to overcome the limitations of MLPs described in the previous Sect. [79]. There are two key ideas behind convolutional networks:

1. statistically meaningful interactions tend to concentrate locally, e.g., within a short time window around an event;
2. shift-invariance (e.g., in time) can be exploited to share weights, thereby reducing the number of parameters in the model.

Convolutional networks are well-suited to applications in which the desired output is a sequence of predictions, e.g., time-varying event detection, and the concepts being modeled derive only from local interactions. In this section, we will describe one-dimensional and two-dimensional convolutional networks. Though the idea generalizes to higher dimensions, these two formulations are the most practically useful in audio applications.

5.4.3.1 One-Dimensional Convolutional Networks

Given an input observation $z \in \mathbb{R}^{T \times d}$, a one-dimensional convolutional filter with coefficients $w \in \mathbb{R}^{n \times d}$ and bias b produces a response $\rho(w * z + b)$, where $w * z$

denotes the “valid”¹⁴ discrete convolution of w with z ¹⁵:

$$(w * z)[t] := \sum_{j=1}^n \langle w[j], z[t + j - \lceil n/2 \rceil] \rangle. \quad (5.54)$$

Here, $n \leq T$ denotes the size of the receptive field of the filter w , j indexes the filter coefficients and position within the input signal, and d indicates the dimensionality (number of channels) in the input. By convention, the receptive field n is usually chosen to be an odd number so that responses are centered around an observation.

A convolutional layer $f_i : \mathbb{R}^{T_{i-1} \times d_{i-1}} \rightarrow \mathbb{R}^{(T_{i-1} - n_i + 1) \times d_i}$ consists of d_i convolutional filters, collectively denoted as w_i , and bias terms (collected as b_i):

$$f_i(z | \theta) := \rho_i(w_i * z + b_i). \quad (5.55)$$

The output of f_i , sometimes called a *feature map* is of slightly reduced extent than the input (due to valid-mode convolution), and can be interpreted as sliding an MLP with weights $w \in \mathbb{R}^{d_{i-1} \times d_i}$ over every position in the input z . An example of this architecture is illustrated in Fig. 5.6.

Note that although the input and output of a convolutional layer are two-dimensional, the first dimension is assumed to encode “temporal position” (over which the convolution ranges) and the second dimension encodes (unordered) filter channel responses as a function of position. When the input has only a single observation channel (e.g., waveform amplitude), which is represented as $x \in \mathbb{R}^{T_0 \times 1}$. For higher-dimensional input—e.g., spectrograms— d_0 corresponds to the number of observed features at each time step (e.g., number of frequency bins).

Cascading convolutional layers can be interpreted as providing hierarchical representations. However, in the form given above, the receptive field of the i th layer’s filter is linear in i . *Pooling layers* down-sample feature maps between convolutional layers, so that deeper layers effectively integrate larger extents of data. A one-dimensional pooling layer has two parameters: width r and stride s :

$$f_i(z | \theta)[t] := \text{agg} \left(z[ts + j] \mid j \in \left[-\left\lfloor \frac{r}{2} \right\rfloor, \left\lfloor \frac{r}{2} \right\rfloor \right] \right), \quad (5.56)$$

where agg denotes an aggregation operator, such as $\max()$ or $\text{sum}()$, and is applied independently to each channel. Max-pooling is particularly common, as it can be interpreted as a softened version of a logical-or, indicating whether filter had positive response anywhere within the window around z_{ts} . Usually, the pooling stride is set to match the width ($s = r$), so that there is minimal redundancy in the output of the pooling layer, and the result is a downsampling by a factor of s .

¹⁴A *valid-mode* convolution is one in which the response is computed only at positions where the signal z and filter w fully overlap. For $z \in \mathbb{R}^T$ and $w \in \mathbb{R}^n$, the valid convolution $w * z \in \mathbb{R}^{T-n+1}$.

¹⁵Technically, (5.54) is written as a cross-correlation and not a convolution. However, since the weights w are variables to be learned, and all quantities are real-valued, the distinction is not important.

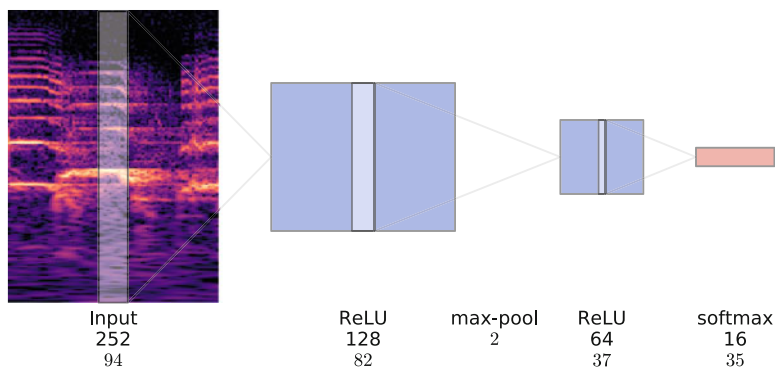


Fig. 5.6 An example of a one-dimensional convolutional network using a spectrogram as input; convolution is performed only over the time dimension (horizontal axis). The vertical axis corresponds to the dimension of each layer. The *shaded regions* indicate the effective receptive field of a single filter in the subsequent layer at the center position. This network takes input in $\mathbb{R}^{T \times 252}$ (in this example, $T = 94$ frames), and applies: $d_1 = 128$ convolutional filters ($n_1 = 13$ frames) with ReLU activation, a downsampling of $r = 2$; $d_3 = 64$ convolutional filters ($n_3 = 5$ frames) with ReLU activation, and finally a convolutional soft-max layer ($n_4 = 3$ frames) mapping to $d_4 = 16$ classes

Typical convolutional architectures alternate convolution layers with pooling layers, which ultimately results in an output layer of shape $T_m \times d_m$, where $T_m < T_0$ is the result of successive pooling and valid-mode convolution operations. Note that T_m is generally a function of T_0 , and will differ for inputs of different length. Care should be taken during training to align the sampling rate of labels y to that of the model's output layer f_m , but this is usually a simple task (e.g., downsampling y).

However, when the desired output exists only at the recording level, then some form of aggregation is required so that the output layer's shape conforms to that of the labels. The two most common approaches to reconciling output shapes are

1. use *global* pooling operators across the convolutional dimension—e.g., max over the entire time axis—followed by a standard MLP architecture; or
2. pass the convolutional outputs directly into fully connected MLP layers (Sect. 5.4.5).

Note that for the global pooling approach to work in general, the model must be able to accommodate input of variable length; this is a common enough operation that most software implementations support global pooling operators. The second approach implicitly limits the model to operate on fixed-dimensional inputs, even at test time. This can make the resulting model inconvenient to deploy, since observation windows must be manually extracted and passed through the model, but the models tend to perform well in practice because they preserve temporal relations among feature activations within the observation.

One-dimensional convolutional networks are often applied to spectrogram representations of audio [65, 80, 116]. While this approach is often successful, it is worth

noting that it cannot learn filters which are invariant to frequency transposition. As a result, it may require a large number of (redundant) first-layer filters to accurately model phenomena which move in frequency.

5.4.3.2 Two-Dimensional Convolutional Networks

The one-dimensional convolutional method described in the previous section generalizes to higher-dimensional data, where multiple dimensions possess a proper notion of locality and ordering. Two-dimensional convolutional architectures are especially common, due to their natural application in image processing [79], which can in turn be adapted to time-frequency representations of audio. The benefits of two-dimensional convolutional architectures on time-frequency representations include a larger effective observation window (due to temporal framing, as in one-dimensional convolutional networks), the ability to leverage frequency decompositions to more clearly locate structures of interest, and the potential for learning representations which are invariant to frequency transposition.

Technically, two-dimensional convolutional layers look much the same as their one-dimensional counterparts. An input observation is now represented as a three-dimensional array $z \in \mathbb{R}^{T \times U \times d}$ where T and U denote temporal and spatial extents, and d denotes non-convolutional input channels. The filter coefficients similarly form a three-dimensional array $w \in \mathbb{R}^{n \times p \times d}$, and the convolution operator $w * z$ is accordingly generalized:

$$(w * z)[t, u] := \sum_{j=1}^n \sum_{k=1}^p \langle w[j, k], z[t + j - \lceil n/2 \rceil, u + k - \lceil p/2 \rceil] \rangle. \quad (5.57)$$

The corresponding layer $f_i : \mathbb{R}^{T_{i-1} \times U_{i-1} \times d_{i-1}} \rightarrow \mathbb{R}^{T_i \times U_i \times d_i}$ otherwise operates analogously to the one-dimensional case (5.55), and pooling operators generalize in a similarly straightforward fashion. For a spectrogram-like input $x \in \mathbb{R}^{T_0 \times U_0 \times d_0}$, we take T_0 to be the number of frames, U_0 the number of frequency bins, and $d_0 = 1$ to indicate the number of channels. Note that larger values of d_0 are possible, if, for instance, one wished to jointly model stereo inputs ($d_0 = 2$, for left and right channels), or some other time- and frequency-synchronous multi-channel representation.

Two-dimensional convolutional networks can be used to learn small, localized filters in the first layer, which can move both vertically (in frequency) and horizontally (in time) [67, 102]. Unlike one-dimensional convolutions, two-dimensional convolution is only well-motivated when the input representation uses a log-scaled frequency representation (e.g., a constant-Q transform), so that the ratio of frequencies covered by a filter of height p bins remains constant regardless of its position.

An example of this architecture is illustrated in Fig. 5.7. The first layer filters in this kind of architecture tend to learn simple, local features, such as transients or sustained tones, which can then be integrated across large frequency ranges by subsequent layers in the network.

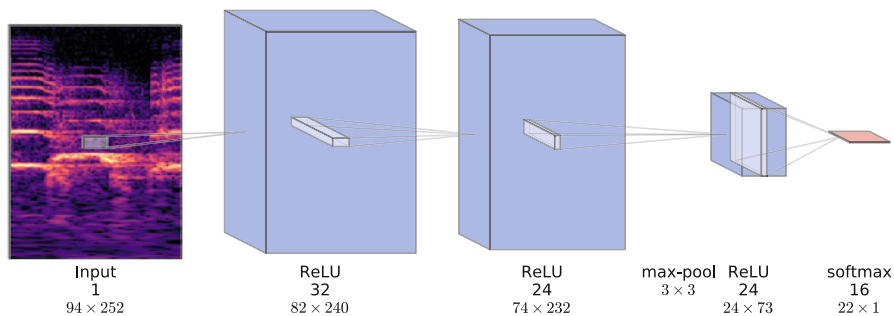


Fig. 5.7 An example of a two-dimensional convolutional network with spectrogram input and local filters. The depth axis corresponds to the dimensionality of each layer, and both horizontal and vertical dimensions are convolutional. This network takes input $x \in \mathbb{R}^{T \times U \times 1}$, and applies the following operations: $d_1 = 32$ convolutional filters (13 frames by 13 frequency bins) with ReLU activations, $d_2 = 24$ convolutional filters (9×9), 3×3 max pooling, $d_4 = 24$ convolutional filters (1×5), and a softmax output layer (all 73 vertical positions) over 16 classes

When the desired output of the model is a time-varying prediction, it is common to introduce a full-height layer (i.e., $p_k = U_{k-1}$), so that all subsequent layers effectively become one-dimensional convolutions over the time dimension. Just as with the one-dimensional case (Sect. 5.4.3.1), global pooling or hybrid architectures (Sect. 5.4.5) can be used in settings that call for fixed-dimensional outputs.

5.4.4 Recurrent Networks

The convolutional networks described in the previous section are effective at modeling short-range interactions, due to their limited spatial locality. While pooling operators can expand the receptive field of convolutional filters, they are still not well-suited to modeling long-range interactions, or interactions with variable-length dependencies, which are common in certain forms of audio (e.g., spoken language or music). Recurrent networks [43, 100, 120] provide a more flexible framework in which to model sequential data. Rather than modeling a finite receptive field, observations are encoded and accumulated over temporal or spatial dimensions as latent state variables.

5.4.4.1 Recursive Networks

Like MLPs and convolutional networks, recurrent networks—also called recurrent neural networks (RNNs)—are built up by layers of processing units. In the simplest generic form, a recurrent layer defines a *state vector* $h[t] \in \mathbb{R}^{d_i}$ (at time index t), which is driven by input $z[t] \in \mathbb{R}^{d_{i-1}}$ and the state vector at the previous time $h[t-1]$:

$$h[t] := \rho(w^T z[t] + v^T h[t-1] + b), \quad (5.58)$$

where the layer parameters $\theta = (w, v, b)$ consist of *input weights* $w \in \mathbb{R}^{d_{i-1} \times d_i}$, *recurrent weights* $v \in \mathbb{R}^{d_i \times d_i}$, bias vector $b \in \mathbb{R}^{d_i}$, and element-wise non-linearity ρ . The model is *recurrent* because the state at time t depends on the state at time $t - 1$ (which in turn depends on $t - 2$, and so on) through the recurrent weights v , which play a role similar to the transition matrix in a hidden Markov model (5.37).¹⁶ A recurrent layer therefore integrates information over all $t' \leq t$ to produce the output state vector h_t at time t , and can thus be used to model sequential data with variable-length dependencies. The initial hidden state $h[0]$ is typically set to the all-zeros vector, so that $h[1] = \rho(w^T z[1] + b)$ is driven only by the input and bias.

Just as with MLPs or convolutional networks, recursive networks can be stacked in a hierarchy of layers. The output of a recurrent layer $f_i : \mathbb{R}^{T \times d_{i-1}} \rightarrow \mathbb{R}^{T \times d_i}$ is the sequence of state vectors:

$$f_i(z | \theta) := (h[t])_{t=1}^T, \quad (5.59)$$

which can in turn be used as inputs to a second recursive layer, or to a convolutional layer which maps the hidden state vectors $h[t]$ to predicted outputs.

Learning the weights w and v for a recurrent layer is computationally challenging, since the gradient calculation depends on the entire state sequence. The standard practical approach to this problem is *back-propagation through time* (BPTT) [121], which approximates the full gradient by unrolling (5.58) up to a finite number k of time steps, and applying standard back-propagation to estimate gradients over length- k sub-sequences of the input. The BPTT approach for standard recurrent networks is known to suffer from the *vanishing and exploding gradient problem*, due to the cumulative effect of iteratively applying the state transformation v [10, 90]. In practice, this can limit the applicability of the recurrent formulation defined in (5.58) to relatively short sequences, though attempts have been made to apply the method to sequential tasks such as musical chord recognition [19] or phoneme sequence modeling [20]. For a comprehensive introduction to recursive networks and their challenges, we refer readers to Graves [56].

5.4.4.2 Gated Recurrent Units

The recently proposed gated recurrent unit (GRU) [25] architecture was explicitly designed to mitigate the challenges of gradient-based training of recurrent networks described in the previous section. Although the GRU architecture was proposed as a simplification of the long short-term memory (LSTM) architecture (Sect. 5.4.4.3), we present it here first to ease exposition.

Formally, a GRU consists of a *reset vector* $r[t] \in \mathbb{R}^{d_i}$ and an *update vector* $u[t] \in \mathbb{R}^{d_i}$, in addition to the hidden state vector $h[t] \in \mathbb{R}^{d_i}$. The state equations

¹⁶A key distinction between recurrent networks and HMMs is that the “state space” in a recurrent network is continuous, i.e., $h_t \in \mathbb{R}^{d_i}$.

are defined as follows:

$$r[t] := \sigma(w_r^\top z[t] + v_r^\top h[t-1] + b_r) \quad (5.60a)$$

$$u[t] := \sigma(w_u^\top z[t] + v_u^\top h[t-1] + b_u) \quad (5.60b)$$

$$\hat{h}[t] := \rho(w_h^\top z[t] + v_h^\top (r[t] \odot h[t-1]) + b_h) \quad (5.60c)$$

$$h[t] := u[t] \odot h[t-1] + (1 - u[t]) \odot \hat{h}[t], \quad (5.60d)$$

where \odot denotes the element-wise (Hadamard) product, σ is the logistic function, and the weights $w_r, w_u, w_h \in \mathbb{R}^{d_{i-1} \times d_i}$ and $v_u, v_r, v_h \in \mathbb{R}^{d_i \times d_i}$ and biases $b_r, b_u, b_h \in \mathbb{R}^{d_i}$ are all defined analogously to the standard recurrent layer (5.58). The transfer function ρ in (5.60c) is typically taken to be tanh. Non-saturating transfer functions are discouraged for this setting because they allow $h[t]$, and thus $v^\top h[t-1]$ to grow without bound, which in turn causes both exploding gradients (on v weights) and can limit the influence of the inputs $z[t]$ in the update equations.

The *gate* variables $r[t]$ and $u[t]$ control the updates to the state vector $h[t]$, which is a convex combination of the previous state $h[t-1]$ and a proposed next state $\hat{h}[t]$. When the update vector $u[t]$ is close to 1, (5.60d) persists the previous state $h[t-1]$ and discards the proposed state $\hat{h}[t]$. When $u[t]$ is close to 0 and $r[t]$ is close to 1, (5.60d) updates $h[t]$ according to the standard recurrent layer equation (5.58). When both $u[t]$ and $r[t]$ are close to 0, $h[t]$ “resets” to $\rho(w_h^\top z[t] + b_h)$, as if $z[t]$ was the first observation in the sequence. As in (5.59), the output of a GRU layer is the concatenation of hidden state vectors $(h[t])_{t=1}^T$.

Like a standard recurrent network, GRUs are also trained using the BPTT method. However, because a GRU can persist state vectors across long extents—when u_t stays near 1—the hidden state h_t does not result directly from successive applications of the transformation matrix v , so it is less susceptible to the vanishing/exploding gradient problem. Similarly, the reset variables allow the GRU to discard state information once it is no longer needed. Consequently, GRUs have been demonstrated to perform well for long-range sequence modeling tasks, such as machine translation [26].

5.4.4.3 Long Short-Term Memory Networks

Long short-term memory (LSTM) networks [66] were proposed long before the gated recurrent unit model of the previous section, but are substantially more complicated. Nonetheless, the LSTM architecture has been demonstrated to be effective for modeling long-range sequential data [112].

An LSTM layer consists of three *gate* vectors: the *input gate* $i[t]$, the *forget gate* $f[t]$, and the *output gate* $o[t]$, as well as the *memory cell* $c[t]$, and the state vector $h[t]$.

Following the formulation of Graves [57], the updates are defined as follows¹⁷:

$$i[t] := \sigma(w_i^\top z[t] + v_i^\top h[t-1] + b_i) \quad (5.61a)$$

$$f[t] := \sigma(w_f^\top z[t] + v_f^\top h[t-1] + b_f) \quad (5.61b)$$

$$o[t] := \sigma(w_o^\top z[t] + v_o^\top h[t-1] + b_o) \quad (5.61c)$$

$$\hat{c}[t] := \rho_c(w_c^\top z[t] + v_c^\top h[t-1] + b_c) \quad (5.61d)$$

$$c[t] := f[t] \odot c[t-1] + i[t] \odot \hat{c}[t] \quad (5.61e)$$

$$h[t] := o[t] \odot \rho_h(c[t]). \quad (5.61f)$$

The memory cell and all gate units have the standard recurrent net parameters $w \in \mathbb{R}^{d_{i-1} \times d_i}$, $v \in \mathbb{R}^{d_i \times d_i}$, $b \in \mathbb{R}^{d_i}$.

Working backward through the equations, the hidden state $h[t]$ (5.61f) can be interpreted as a point-wise non-linear transformation of the memory cell $c[t]$, which has been masked by the output gate $o[t]$. The output gate (5.61c) thus limits which elements of the memory cell are propagated through the recurrent updates (5.61a–5.61c). This is analogous to the reset functionality of a GRU.

The memory cell $c[t]$ (5.61e) behaves similarly to the hidden state $h[t]$ of the GRU (5.60d), except that “update” variable has been decoupled into the *input* and *forget* gates $i[t]$ and $f[t]$. When the forget gate $f[t]$ is low, it masks out elements from the previous memory cell value $c[t-1]$; when the input gate $i[t]$ is high, it integrates the proposed value $\hat{c}[t]$. Because $f[t]$ and $i[t]$ do not necessarily sum to 1, an additional transfer function ρ_h is included in (5.61f) to preserve boundedness of the hidden state vector $h[t]$. As in the GRU, \tanh is the typical choice for the transfer functions ρ_h and ρ_c .

Recently, two empirical studies have studied the importance of the various components of the LSTM architecture [60, 72]. Taken together, their findings indicate that the forget gate $f[t]$ is critical to modeling long-range interactions. In particular, Józefowicz et al. note that it is helpful to initialize the bias term b_f to be relatively large (at least 1) so that the $f[t]$ stays high (propagates previous state) early in training [72]. Both studies also found that across several tasks, the simplified “update” logic of the GRU performs comparably to the more elaborate forget/input logic of the standard LSTM.

¹⁷The presentation of Graves [57] differs slightly in its inclusion of “peephole” connections [51]. We omit these connections here for clarity of presentation, and because recent studies have not demonstrated their efficacy [60].

5.4.4.4 Bi-directional Networks

The RNN, GRU, and LSTM architectures described in the previous sections are all designed to integrate information in one direction along a particular dimension, e.g., forward in time. In some applications, it can be beneficial to integrate information across both directions. Bi-directional recurrent networks (BRNNs) achieve this by a simple reduction to the standard one-directional architecture [104].

A BRNN layer $f_i(z|\theta)$ consists of two standard recurrent layers: the *forward layer* \vec{f}_i and the *backward layer* \overleftarrow{f}_i . The BRNN layer $f_i: \mathbb{R}^{T \times d_{i-1}} \rightarrow \mathbb{R}^{T \times d_i}$ is the concatenation:

$$f_i(z|\theta) := \begin{bmatrix} \vec{f}_i(z|\theta) \\ \text{rev}\left(\overleftarrow{f}_i(\text{rev}(z)|\theta)\right) \end{bmatrix}, \quad (5.62)$$

where $\text{rev}(\cdot)$ reverses its input along the recurrent dimension, and d_i combines the dimensionality of the forward and backward layers.¹⁸ The output $f_i(z|\theta)[t]$ at time t thus includes information integrated across the entire sequence, before and after t . This approach can be easily adapted to LSTM [59] and GRU architectures [6].

Bi-directional networks—in particular, the B-LSTM approach—have been demonstrated to be effective for a wide range of audio analysis tasks, including beat tracking [17], speech recognition [58, 84], and event detection [89]. Unless the application requires forward-sequential processing, e.g., online prediction, bi-directional networks are strictly more powerful, and generally preferred.

5.4.5 Hybrid Architectures

The previous sections covered a range of architectural units for deep networks, but in practice, these architectures are not often used in isolation. Instead, practitioners often design models using combinations of the techniques described above. Here, we briefly summarize the two most commonly used hybrid architectures.

5.4.5.1 Convolutional + Dense

As briefly mentioned in Sect. 5.4.3, many applications consist of variable-length input with fixed-length output, e.g., assigning a single label to an entire audio excerpt. This presents a problem for purely convolutional architectures, where in the absence of global pooling operators, the output length is proportional to the input

¹⁸Some authors define the BRNN output (5.62) as a non-linear transformation of the concatenated state vectors [55]. This formulation is equivalent to (5.62) followed by a one-dimensional convolutional layer with a receptive field $n_i = 1$, so we opt for the simpler definition here.

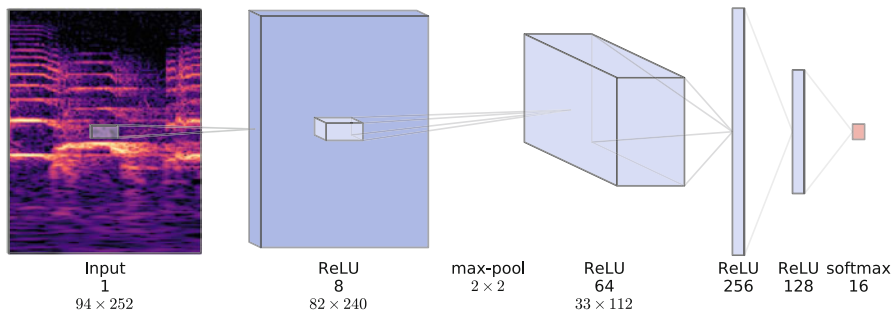


Fig. 5.8 An example of a convolutional-dense architecture. Two convolutional layers are followed by two dense layers and a 16-class output layer

length (or spatial extents, in two-dimensional convolutional models). While global pooling operators can resolve this by aggregating over the entirety of the variable-length input dimensions, the resulting summaries cannot model the dynamics of interactions between features over the convolutional dimension. As a concrete example,

$$\max([0, 1]) = \max([1, 0]) = 1 \quad (5.63)$$

discards the ordering information of the input, which may be relevant for describing certain phenomena.

A common solution to this problem is to replace global pooling operators with dense connections—i.e., one or more MLP layers, also called “fully connected” in this context—to map convolutional outputs to a fixed-dimensional representation. An example of this type of architecture is illustrated in Fig. 5.8.

Note that the convolutional-dense architecture requires all input data x be truncated or padded to a fixed length. Consequently, when deploying the resulting model, the input data must be sliced into fixed-length observation windows, and the resulting window predictions must be collected and aggregated to produce the final prediction. With this approach, care must be taken to ensure that the model evaluation corresponds to quantity of interest (e.g., recording-level rather than window-level prediction). Nonetheless, the general convolutional-dense approach has been demonstrated to perform well on a wide array of tasks [37, 93, 102, 116].

5.4.5.2 Convolutional + Recurrent

Another increasingly common hybrid architecture is to follow one or more convolutional layers by recurrent layers. This approach—alternately known as *convolutional encoder-recurrent decoder*, or *convolutional-recurrent neural network* (CRNN)—combines local feature extraction with global feature integration. Although this architecture is only recently becoming popular, it has been demonstrated to be effective in several applications, including speech recognition [3], image recognition [123], text analysis [113], and music transcription [108].

5.5 Improving Model Stability

In audio applications, the efficacy of a model is often impeded by limited access to a large, well-annotated, and representative sample of training data. Models trained on small, biased, or insufficiently varied data sets can over-fit and generalize poorly to unseen data. Practitioners have developed several techniques to mitigate this issue, and in this section, we briefly summarize three important concepts: data augmentation, domain adaptation, and ensemble methods.

5.5.1 Data Augmentation

Data augmentation is an increasingly popular method to help mitigate sampling bias: the general idea is to perturb the training data during training to inject more variance into the sample. By doing so, the model is exposed to a larger and more varied training sample, and may therefore be better able to characterize the decision boundaries between classes.

Perturbations can range from simple effects like additive background noise, constant pitch shifting, and time stretching [83, 102], to more sophisticated, domain-specific deformations like vocal tract length perturbation [34, 70] or variable speed perturbation [76]. In general, the idea is that training on these modified examples can help the model become invariant to the chosen deformations. Care must be taken to ensure that the deformations applied to the input audio leave the labels unmodified (or at least modified in a controlled way) [83].

5.5.2 Domain Adaptation

Throughout this chapter, there has been an underlying assumption that the training sample S is drawn I.I.D. from the test distribution \mathcal{D} . In practice, this assumption is often violated. Training data can be biased, e.g., when labeled examples produced in one recording environment are used to develop a model which is deployed in a different environment. In general, this problem is known as *domain adaptation* [16]: a model is trained on a sample S drawn from a different domain (distribution) than the eventual target domain.

In general, methods for domain adaptation require access to a labeled training set S drawn from a source distribution \mathcal{D}_s , and an unlabeled sample S' drawn from the target distribution \mathcal{D} . The majority of domain adaptation techniques operate by example weighting. These methods replace the unweighted sum in the learning objective (5.6) by a weighted sum so that it better approximates the loss on the target distribution \mathcal{D} [14, 29, 61]. Alternatively, feature transformation methods distort the training data features so that it is difficult to distinguish samples of \mathcal{D}_s from those

of \mathcal{D} [46, 49, 53]. In both cases, the underlying goal is to minimize the discrepancy between the training distribution (or the loss estimated on the training sample) and the target distribution.

5.5.3 Ensemble Methods

Many of the methods described throughout this chapter can be sensitive to certain modeling decisions, such as input representation, network architecture, initialization schemes, or sampling of training data. As mentioned in the beginning of this chapter, a common remedy is to optimize over these decisions by using withheld validation set. However, this can still bias the resulting model if the validation sets are too small or insufficiently varied.

A complementary approach to combine multiple predictors (h_1, \dots, h_n) in an *ensemble*. There are many ways to go about this, such as majority voting over predictions, or weighted averaging over scores/likelihoods [21, 39]. In practice, when a single model appears to be reaching a performance plateau for a given task, an ensemble can often provide a modest improvement.

5.6 Conclusions and Further Reading

This chapter aimed to provide a high-level overview of supervised machine learning techniques for classification problems. When faced with a specific classification problem, the abundance and diversity of available techniques can present a difficult choice for practitioners. While there is no general recipe for selecting an appropriate method, there are several factors to consider in the process.

The first, and most important factor, is the availability and form of training data. Discriminative models generally require strongly labeled examples, both positively and negatively labeled. For example, in implementing a discriminative bird song detector, it's just as important to provide examples that do not include birds, so that the model can learn to discriminate. If negative examples are not available, a generative, class-conditional model may be more appropriate, but it may require a larger training sample than a discriminative method, due to the increased complexity of modeling the joint density $\mathbf{P}_\theta[x, y]$.

In audio applications, the characteristics of the target concept can also play an important role. Some concepts are obviously localized in time (e.g., transient sound events like gunshots), while others are diffused over long extents (e.g., in scene classification), and others are distinguished by dynamics over intermediate durations (e.g., musical rhythms). These characteristics should be taken into consideration when deciding between localized models (e.g., convolutional networks), dynamic models (HMMs or recurrent networks), or global models that integrate across the entirety of an observation (e.g., the bag-of-frames models described in Chap. 4).

Although machine learning algorithms are generally characterized by the loss functions they attempt to minimize, the treatment presented in this chapter only covers the relatively well-understood binary and multi-class categorization loss functions. In practical applications, the utility of a model can be measured according to a much broader space of evaluation criteria, which are discussed in Chap. 6. Bridging the gap between evaluation and modeling is an area of active research, which broadly falls under the umbrella of *structured output prediction* in the machine learning literature [78, 115].

Additionally, this chapter presents the simplest learning paradigm, in which a fully annotated sample is available for parameter estimation. In reality, a variety of learning paradigms exist, each making different assumptions about the training and test data. These formulations include: *semi-supervised learning* [24], where unlabeled observations are also available; *multiple-instance learning*, where a positive label is applied to a collection of observations, indicating at least one of which is a positive example [40]; and *positive-unlabeled learning*, where labels are only available for a subset of the target concepts, and unlabeled examples may or may not belong to those classes [42]. The choice of learning paradigm ultimately derives from the form of training data available, and how the resulting model will be deployed to make predictions.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: large-scale machine learning on heterogeneous systems (2015). <http://tensorflow.org/>, Software available from tensorflow.org
2. Akaike, H.: Likelihood of a model and information criteria. *J. Econom.* **16**(1), 3–14 (1981)
3. Amodei, D., Anubhai, R., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., Chrzanowski, M., Coates, A., Diamos, G., et al.: Deep speech 2: end-to-end speech recognition in English and Mandarin (2015). arXiv preprint arXiv:1512.02595
4. Andrieu, C., De Freitas, N., Doucet, A., Jordan, M.I.: An introduction to MCMC for machine learning. *Mach. Learn.* **50**(1–2), 5–43 (2003)
5. Antoniak, C.E.: Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Ann. Stat.* **2**, 1152–1174 (1974)
6. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2014). arXiv preprint arXiv:1409.0473
7. Baum, L.E., Petrie, T.: Statistical inference for probabilistic functions of finite state Markov chains. *Ann. Math. Stat.* **37**(6), 1554–1563 (1966)
8. Beal, M.J.: Variational algorithms for approximate Bayesian inference. University of London (2003)
9. Bellet, A., Habrard, A., Sebban, M.: A survey on metric learning for feature vectors and structured data (2013). arXiv preprint arXiv:1306.6709
10. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **5**(2), 157–166 (1994)

11. Bergstra, J., Bastien, F., Breuleux, O., Lamblin, P., Pascanu, R., Delalleau, O., Desjardins, G., Warde-Farley, D., Goodfellow, I., Bergeron, A., et al.: Theano: deep learning on GPUs with python. In: *Big Learn, Neural Information Processing Systems Workshop* (2011)
12. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **13**(Feb), 281–305 (2012)
13. Bergstra, J.S., Bardenet, R., Bengio, Y., Kégl, B.: Algorithms for hyper-parameter optimization. In: *Advances in Neural Information Processing Systems*, pp. 2546–2554 (2011)
14. Bickel, S., Brückner, M., Scheffer, T.: Discriminative learning under covariate shift. *J. Mach. Learn. Res.* **10**(Sep), 2137–2155 (2009)
15. Blei, D.M., Jordan, M.I., et al.: Variational inference for Dirichlet process mixtures. *Bayesian Anal.* **1**(1), 121–144 (2006)
16. Blitzer, J., McDonald, R., Pereira, F.: Domain adaptation with structural correspondence learning. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 120–128. Association for Computational Linguistics, Trento (2006)
17. Böck, S., Schedl, M.: Enhanced beat tracking with context-aware neural networks. In: *Proceedings of the International Conference on Digital Audio Effects* (2011)
18. Bottou, L.: Stochastic gradient learning in neural networks. *Proc. Neuro-Nimes* **91**(8), 687–696 (1991)
19. Boulanger-Lewandowski, N., Bengio, Y., Vincent, P.: Audio chord recognition with recurrent neural networks. In: *Proceedings of the International Conference on Music Information Retrieval*, pp. 335–340. Citeseer (2013)
20. Boulanger-Lewandowski, N., Droppo, J., Seltzer, M., Yu, D.: Phone sequence modeling with recurrent neural networks. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5417–5421. IEEE, New York (2014)
21. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
22. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: *Classification and regression trees*. CRC Press, New York (1984)
23. Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M.A., Guo, J., Li, P., Riddell, A.: Stan: a probabilistic programming language. *J. Stat. Softw.* **20**, 1–37 (2016)
24. Chapelle, O., Scholkopf, B., Zien, A.: Semi-supervised learning. *IEEE Trans. Neural Netw.* **20**(3), 542–542 (2009)
25. Cho, K., van Merriënboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pp. 1724–1734 (2014)
26. Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: encoder-decoder approaches (2014). arXiv preprint arXiv:1409.1259
27. Chollet, F.: Keras. <https://github.com/fchollet/keras> (2015). Retrieved on 2017-01-02.
28. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: a matlab-like environment for machine learning. In: *Big Learn, Neural Information Processing Systems Workshop, EPFL-CONF-192376* (2011)
29. Cortes, C., Mohri, M.: Domain adaptation and sample bias correction theory and algorithm for regression. *Theor. Comput. Sci.* **519**, 103–126 (2014)
30. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
31. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. Wiley, New York (2012)
32. Cox, D.R.: The regression analysis of binary sequences. *J. R. Stat. Soc. Ser. B Methodol.* **20**, 215–242 (1958)
33. Crammer, K., Singer, Y.: On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res.* **2**(Dec), 265–292 (2001)
34. Cui, X., Goel, V., Kingsbury, B.: Data augmentation for deep neural network acoustic modeling. *IEEE/ACM Trans. Audio Speech Lang. Process. (TASLP)* **23**(9), 1469–1477 (2015)

35. Dasarathy, B.V.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1991)
36. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B (Methodological)* **39**, 1–38 (1977)
37. Dieleman, S., Schrauwen, B.: End-to-end learning for music audio. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6964–6968. IEEE, New York (2014)
38. Dieleman, S., Schlüter, J., Raffel, C., Olson, E., Sønderby, S.K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J.D., Heilman, M., Diogo149, McFee, B., Weideman, H., Takacs84, Peterderivaz, Jon, Instagibbs, Rasul, D.K., CongLiu, Britefury, Degraive, J.: Lasagne: first release (2015). doi:10.5281/zenodo.27878. <https://doi.org/10.5281/zenodo.27878>
39. Dietterich, T.G.: Ensemble learning. In: *The Handbook of Brain Theory and Neural Networks*, 2nd edn., pp. 110–125. MIT Press, Cambridge, MA (2002)
40. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.* **89**(1), 31–71 (1997)
41. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **12**(Jul), 2121–2159 (2011)
42. Elkan, C., Noto, K.: Learning classifiers from only positive and unlabeled data. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 213–220. ACM, New York (2008)
43. Elman, J.L.: Finding structure in time. *Cogn. Sci.* **14**(2), 179–211 (1990)
44. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: a library for large linear classification. *J. Mach. Learn. Res.* **9**(Aug), 1871–1874 (2008)
45. Feldman, V., Guruswami, V., Raghavendra, P., Wu, Y.: Agnostic learning of monomials by halfspaces is hard. In: *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pp. 385–394. IEEE Computer Society, New York (2009)
46. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2960–2967 (2013)
47. Fix, E., Hodges, J.L. Jr.: Discriminatory analysis-nonparametric discrimination: consistency properties. Technical Report, DTIC Document (1951)
48. Friedman, J., Hastie, T., Tibshirani, R.: *The Elements of Statistical Learning*. Springer Series in Statistics, vol. 1. Springer, Berlin (2001)
49. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(59), 1–35 (2016). <http://jmlr.org/papers/v17/15-239.html>
50. Gelfand, A.E., Smith, A.F.: Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* **85**(410), 398–409 (1990)
51. Gers, F.A., Schmidhuber, J.: Recurrent nets that time and count. In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, vol. 3, pp. 189–194. IEEE, New York (2000)
52. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: *International Conference on Artificial Intelligence and Statistics*, vol. 9, pp. 249–256 (2010)
53. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2066–2073 (2012)
54. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press, Cambridge, MA (2016). <http://www.deeplearningbook.org>
55. Graves, A.: Sequence transduction with recurrent neural networks. CoRR abs/1211.3711 (2012). <http://arxiv.org/abs/1211.3711>
56. Graves, A.: *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, Berlin (2012)

57. Graves, A.: Generating sequences with recurrent neural networks (2013). arXiv preprint arXiv:1308.0850
58. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: Proceedings of the International Conference on Machine Learning, vol. 14, pp. 1764–1772 (2014)
59. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **18**(5), 602–610 (2005)
60. Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., Schmidhuber, J.: LSTM: a search space odyssey (2015). arXiv preprint arXiv:1503.04069
61. Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., Schölkopf, B.: Covariate shift by kernel mean matching. *Dataset Shift Mach. Learn.* **3**(4), 5 (2009)
62. Hastings, W.K.: Monte carlo sampling methods using Markov chains and their applications. *Biometrika* **57**(1), 97–109 (1970)
63. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)
64. Heittola, T., Mesáros, A., Eronen, A., Virtanen, T.: Context-dependent sound event detection. *EURASIP J. Audio Speech Music Process.* **2013**(1), 1–13 (2013)
65. Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., et al.: Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Process. Mag.* **29**(6), 82–97 (2012)
66. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
67. Humphrey, E.J., Bello, J.P.: Rethinking automatic chord recognition with convolutional neural networks. In: 2012 11th International Conference on Machine Learning and Applications (ICMLA), vol. 2, pp. 357–362. IEEE, New York (2012)
68. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pp. 448–456 (2015)
69. Ishwaran, H., Zarepour, M.: Exact and approximate sum representations for the Dirichlet process. *Can. J. Stat.* **30**(2), 269–283 (2002)
70. Jaitly, N., Hinton, G.E.: Vocal tract length perturbation (VTLP) improves speech recognition. In: Proceedings of ICML Workshop on Deep Learning for Audio, Speech and Language (2013)
71. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: convolutional architecture for fast feature embedding (2014). arXiv preprint arXiv:1408.5093
72. Józefowicz, R., Zaremba, W., Sutskever, I.: An empirical exploration of recurrent network architectures. In: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, 6–11 July 2015, pp. 2342–2350 (2015). <http://jmlr.org/proceedings/papers/v37/jozefowicz15.html>
73. Jurafsky, D., Martin, J.H.: Speech and language processing: an introduction to speech recognition. Computational Linguistics and Natural Language Processing. Prentice Hall, Upper Saddle River (2008)
74. Kearns, M.J.: The Computational Complexity of Machine Learning. MIT Press, Cambridge (1990)
75. Kingma, D., Ba, J.: Adam: a method for stochastic optimization (2014). arXiv preprint arXiv:1412.6980
76. Ko, T., Peddinti, V., Povey, D., Khudanpur, S.: Audio augmentation for speech recognition. In: Proceedings of INTERSPEECH (2015)
77. Kullback, S., Leibler, R.A.: On information and sufficiency. *Ann. Math. Stat.* **22**(1), 79–86 (1951)

78. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the International Conference on Machine Learning, ICML, vol. 1, pp. 282–289 (2001)
79. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
80. Lee, H., Pham, P., Largman, Y., Ng, A.Y.: Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Advances in Neural Information Processing Systems, pp. 1096–1104 (2009)
81. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of ICML Workshop on Deep Learning for Audio, Speech, and Language Processing (2013)
82. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Oakland, vol. 1, no. 14, pp. 281–297 (1967)
83. McFee, B., Humphrey, E.J., Bello, J.P.: A software framework for musical data augmentation. In: International Society for Music Information Retrieval Conference (ISMIR) (2015)
84. Mesnil, G., He, X., Deng, L., Bengio, Y.: Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In: Proceedings of INTERSPEECH, pp. 3771–3775 (2013)
85. Murphy, K.P.: *Machine Learning: A Probabilistic Perspective*. MIT Press, Cambridge (2012)
86. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 807–814 (2010)
87. Neal, R.M.: Probabilistic inference using Markov chain monte carlo methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto, Toronto, Ontario (1993)
88. Nesterov, Y.: A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Sov. Math. Dokl.* **27**(2), 372–376 (1983)
89. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6440–6444. IEEE, New York (2016)
90. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: Proceedings of the 30th International Conference on Machine Learning, ICML (3), vol. 28, pp. 1310–1318 (2013)
91. Pearson, K.: Contributions to the mathematical theory of evolution. *Philos. Trans. R. Soc. Lond. A* **185**, 71–110 (1894)
92. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* **12**(Oct), 2825–2830 (2011)
93. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6. IEEE, New York (2015)
94. Platt, J., et al.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.* **10**(3), 61–74 (1999)
95. Polyak, B.T.: Some methods of speeding up the convergence of iteration methods. *USSR Comput. Math. Math. Phys.* **4**(5), 1–17 (1964)
96. Rabiner, L., Juang, B.: An introduction to hidden Markov models. *IEEE ASSP Mag.* **3**(1), 4–16 (1986)
97. Raiffa, H.: *Bayesian decision theory. Recent Developments in Information and Decision Processes*, pp. 92–101. Macmillan, New York (1962)
98. Rasmussen, C.E.: The infinite Gaussian mixture model. In: *Neural Information Processing Systems*, vol. 12, pp. 554–560 (1999)
99. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **65**(6), 386 (1958)

100. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. *Cogn. Model.* **5**(3), 1 (1988)
101. Rumelhart, D.E., McClelland, J.L., Group, P.R., et al.: *Parallel Distributed Processing*, vol. 1. IEEE, New York (1988)
102. Schlüter, J., Grill, T.: Exploring data augmentation for improved singing voice detection with neural networks. In: 16th International Society for Music Information Retrieval Conference (ISMIR-2015) (2015)
103. Schölkopf, B., Herbrich, R., Smola, A.J.: A generalized representer theorem. In: International Conference on Computational Learning Theory, pp. 416–426. Springer, London (2001)
104. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **45**(11), 2673–2681 (1997)
105. Schwarz, G., et al.: Estimating the dimension of a model. *Ann. Stat.* **6**(2), 461–464 (1978)
106. Shalev-Shwartz, S., Ben-David, S.: *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, Cambridge (2014)
107. Shawe-Taylor, J., Cristianini, N.: *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge (2004)
108. Sigtia, S., Benetos, E., Dixon, S.: An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(5), 927–939 (2016)
109. Sjöberg, J., Ljung, L.: Overtraining, regularization and searching for a minimum, with application to neural networks. *Int. J. Control.* **62**(6), 1391–1407 (1995)
110. Snoek, J., Larochelle, H., Adams, R.P.: Practical Bayesian optimization of machine learning algorithms. In: *Advances in Neural Information Processing Systems*, pp. 2951–2959 (2012)
111. Sutskever, I., Martens, J., Dahl, G.E., Hinton, G.E.: On the importance of initialization and momentum in deep learning. In: *Proceedings of the International Conference on International Conference on Machine Learning, ICML (3)*, vol. 28, pp. 1139–1147 (2013)
112. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: *Advances in Neural Information Processing Systems*, pp. 3104–3112 (2014)
113. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1422–1432 (2015)
114. Tran, D., Kucukelbir, A., Dieng, A.B., Rudolph, M., Liang, D., Blei, D.M.: Edward: a library for probabilistic modeling, inference, and criticism (2016). arXiv preprint arXiv:1610.09787
115. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.* **6**(Sep), 1453–1484 (2005)
116. Van den Oord, A., Dieleman, S., Schrauwen, B.: Deep content-based music recommendation. In: *Advances in Neural Information Processing Systems*, pp. 2643–2651 (2013)
117. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inf. Theory* **13**(2), 260–269 (1967)
118. Wainwright, M.J., Jordan, M.I.: Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.* **1**(1–2), 1–305 (2008)
119. Watanabe, S.: Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *J. Mach. Learn. Res.* **11**(Dec), 3571–3594 (2010)
120. Werbos, P.J.: Generalization of backpropagation with application to a recurrent gas market model. *Neural Netw.* **1**(4), 339–356 (1988)
121. Werbos, P.J.: Backpropagation through time: what it does and how to do it. *Proc. IEEE* **78**(10), 1550–1560 (1990)
122. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 694–699. ACM, New York (2002)
123. Zuo, Z., Shuai, B., Wang, G., Liu, X., Wang, X., Wang, B., Chen, Y.: Convolutional recurrent neural networks: learning spatial dependencies for image representation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 18–26 (2015)

Chapter 6

Datasets and Evaluation

Annamaria Mesaros, Toni Heittola, and Dan Ellis

Abstract Developing computational systems requires methods for evaluating their performance to guide development and compare alternate approaches. A reliable evaluation procedure for a classification or recognition system will involve a standard dataset of example input data along with the intended target output, and well-defined metrics to compare the systems' outputs with this ground truth. This chapter examines the important factors in the design and construction of evaluation datasets and goes through the metrics commonly used in system evaluation, comparing their properties. We include a survey of currently available datasets for environmental sound scene and event recognition and conclude with advice for designing evaluation protocols.

Keywords Audio datasets • Reference annotation • Sound scene labels • Sound event labels • Manual audio annotation • Annotation process design • Evaluation setup • Evaluation metrics • Evaluation protocol • Event-based metrics • Segment-based metrics

6.1 Introduction

Systematic engineering methodology involves quantifiable goals and accurate ways to measure progress towards them. In the context of developing systems for computational analysis of sound scenes and events, we often adopt goals in terms of specific performance metrics to be evaluated on carefully prepared evaluation datasets. Well-constructed evaluations can clarify and guide research, and allow direct comparisons between different approaches, whereas poorly chosen

A. Mesaros (✉) • T. Heittola
Tampere University of Technology, P.O. Box 527, FI-33101 Tampere, Finland
e-mail: annamaria.mesaros@tut.fi; toni.heittola@tut.fi

D. Ellis
Google Inc, 111 8th Ave, New York, NY 10027, USA
e-mail: dpwe@google.com

evaluations can confuse and misdirect a whole field. In this chapter, we examine the considerations in choosing or constructing evaluation datasets, and we examine the different metrics relevant for use in sound scene and event evaluations.

Systematic evaluations have been a central part of several research communities, most notably the automatic speech recognition evaluations pioneered by NIST as part of DARPA's speech recognition programs from the mid-1980s onward [34]. For more than three decades, the speech recognition community has coalesced around a single performance metric—word error rate or WER—and a series of transcribed audio datasets of steadily increasing difficulty, ranging from isolated digits through to real, informal telephone conversations. While it is not uncommon to hear grumbles about the tyranny of WER evaluations, and while very small marginal improvements are sometimes ridiculed, the unquestionable success and impact of speech recognition is testament to the success of this regime in driving steady, cumulative improvements in speech recognition technology.

In some ways it is remarkable that speech recognition has been able to use a single evaluation paradigm for so long. A more common experience is to have no clear, single choice for performance metric, and instead to need to consider and customize metrics for each specific project or application. Speech recognition was also fortunate in having, for many years, a central sponsor willing to fund the significant expense of preparing high-quality, relevant evaluation datasets. From DARPA's point of view, this was a rational investment to ensure the money spent funding research was demonstrably well spent, but for less centrally organized fields it can be difficult for any single research group to have the resources to construct the kind of high-quality, large-scale evaluation dataset and tools needed to promote research progress.

In the following sections, we discuss specific properties and issues relating to the datasets and metrics relevant to sound source and event recognition. Although the field is at present too diverse to easily be served by a single evaluation process, we hope to make clear the different qualities that evaluation procedures and datasets should possess. We will conclude with specific advice on the choice of data and metrics.

6.2 Properties of Audio and Labels

Labeled data has a crucial influence on algorithm development and evaluation in research fields dealing with classification and detection. Any machine learning algorithm is only as good as the data behind it in terms of modeling and generalization properties. Well-established and easily accessible benchmark databases attract the interest of the research community as readily available support for research, thus accelerating the pace of development for related fields. There are many well-known databases in related research areas, such as TIMIT [17] and WSJ [36], for speech recognition or RWC [20] and the Million Song Dataset [7] for various tasks in music information retrieval. In view of this critical influence, the process of creating

a dataset for research is, naturally, very delicate. The content must be carefully selected to provide sufficient coverage of the aspects of interest, sufficient variability in characterizing these aspects, and a sufficient quantity of examples for robust modeling. Unfortunately, there is no rule on what “sufficient” means, as it usually depends on the projected use of the data.

Compared to speech, which has a relatively small, closed set of different categories (phonemes) and rules for how they can follow each other (language model) or music that follows certain rules (notes in certain frequency ratios forming chords, key and pitch evolution during a musical piece), the categories and sequences of environmental sounds are not so straightforward to define, as any object or being may produce a sound. Sounds can be organized into categories based on different properties, such as the sound source (e.g., *cup*), its production mechanism (e.g., *hit*), or even the source properties (e.g., *metal*), and the same sound can belong to many different categories, depending on the chosen property (sound when hitting a metallic cup). In addition, there are no specific rules on how environmental sounds co-occur. For this reason, building a database for environmental sounds is a complicated task, with the choice of classes usually dictated by the targeted research, and the size limited by practical aspects of data collection.

An important component of a dataset is the annotation—the link between the data content (audio, image, or anything else) and the formal categories that associate this content with its meaning for humans. These are the categories to be learned and recognized, and they define the tasks and uses of the data. The textual labels used in annotation must provide a good description of the associated category and not allow misinterpretation. The properties of audio and labels in a dataset are closely interconnected, and it is difficult to discuss one without the other. Also, there is a close relationship between the properties of a dataset and the type of application for which it is meant or can be used for.

We now look at these two components, audio, and labels, in more detail.

6.2.1 Audio Content

From the machine learning perspective, the properties of the audio data facilitate model robustness: The audio must represent the modeled phenomena such that the models learned from it will represent the variability in the acoustic properties expected in the application. As presented in Chap. 5, during training the features extracted from audio are used to create acoustic models for the target categories. Models learned from suitable audio data will be robust and generalize well, resulting in good performance in test situations. To support good representation in machine learning, the audio content of a database should have the following properties:

- *Coverage*: The dataset should contain as many different categories as are relevant to the task (perhaps including future evolutions).

- *Variability*: For each category, there should be examples with variable conditions of generation, recording, etc.
- *Size*: For each category there should be sufficiently many examples.

These three properties are inter-related: For the necessary categories to be well modeled, it is required to have many examples from many different acoustic conditions. Comparable requirements have been identified for selection of suitable sounds for environmental sound research in human perception [51], presented as practical considerations in building a representative set of audio examples for a particular experimental task. Similarly, the selection of a representative set of audio examples is important for machine learning algorithms. In addition, current algorithm development is closely focused on deep learning methods, which continue to benefit from expanded training sets seemingly without limit. A property related to data size is the balance of the amount of data between categories, which is usually necessary to ensure no class is under-represented in the training process [23]. However, a large training dataset is not sufficient to achieve high performance if the training examples fail to span the variability exhibited in the test data and application.

6.2.1.1 Sound Scene Audio Data

A sound scene is the audio environment as recorded in a specific physical or social context, e.g., in a park, in the office, or during a meeting. The sound scene represents the general acoustic circumstances of the given situation; it does not refer to individual sounds, but to the combination of them seen as a whole.

For sound scene classification, coverage in audio content means that the dataset must contain all the scene categories which have to be represented in the task. This is somewhat obvious in closed-set classification problems, where a test audio example is meant to be classified as belonging to one of the existing categories. However, in some cases it is necessary to detect if a test example belongs to a known or unknown category—in the so-called open-set problem. In this situation, there is a need for audio content to represent the unknown category, and the considerations of coverage must still apply.

For each category, audio examples from many different locations are necessary in order to achieve sufficient acoustic variability. This means recordings from different streets, different offices, different restaurants, etc., with different acoustic conditions (weather, time of day, number of people at the scene, . . .). Such variety will support generalization in learning the characteristics of the acoustic scene class. If, however, the goal is to recognize a specific location, only variability in recording conditions is needed. Size-wise, a large number of examples for each category and for each recording condition are desired, aiming so that no case is under-represented.

6.2.1.2 Sound Events Audio Data

A sound event is a sound that we perceive as a separate, individual entity that we can name and recognize, and, conversely, whose acoustic characteristics we can recall based on its description. Sound events are the building blocks of a sound scene and facilitate its interpretation, as explained in Chap. 3.

The presence of all categories relevant to a sound event detection task will provide the required coverage, making a dataset suitable for the given task. In an open-set problem, the test data can belong to a category that was not encountered in training or is not of interest in the task, usually marked as “unknown”; the choice of audio content for this category depends mostly on the expected test situation; sounds that are extremely unlikely to be encountered provide limited value to the dataset.

Compared to other types of audio, sound events generally have a very high acoustic variability. For example, the words in speech exhibit acoustic variability due to aspects like intonation, accent, speaker identity, and mood, but they always have the same underlying structure of phonemes that is employed when building acoustic models. Some sound categories do exhibit structure, for example, footsteps that are composed of individual sounds of a foot tapping the ground, therefore having a temporal structure which can differ due to walking speed, surface, etc. Other sound categories are very broad, and a structure is noticeable only in the subcategories, for example, birdsong being similar only for the same species, possibly requiring a different and more detailed labeling scheme.

Variability for a dataset of sound events can be achieved by having multiple sound instances from the same sound source, recorded in different acoustic conditions (similar to examples of speech from the same person in different situations), and also examples from multiple sound sources that belong to the same category (similar to collecting speech from multiple persons). The acoustic variability of the specific sound sources can be considered as a factor: Variability of natural sound sources is much greater than of non-natural sources (the same dog will bark in many different ways but a given fire alarm will always make the same sound); therefore, for some sound categories it may be unnecessary to have many examples with the exact same acoustic content. Due to the high acoustic variability of sound events, it is impossible to obtain examples of all categories in all conditions; therefore, the acceptable size and variability for a training dataset depends strongly on the expected variability of the application as represented by the test data.

6.2.2 Textual Labels

In supervised learning, the labels are a critical component of the database and in many cases they drive the dataset design. The choice of target application typically dictates the need for specific categories of sounds, which in turn defines the data collection process in terms of content and diversity.

The labels attached to the training data define the categories to be learned; on the test data, the labels dictate the desired output against which to evaluate recognition performance. Unsupervised learning methods can take advantage of unlabeled training data, but labeled data is always necessary for objective evaluation of performance. In consequence, the labels should be representative of natural grouping of phenomena, while also being explicit for the end user, to provide an intuitive understanding of the modeled problem. For this, the labels should have the following properties:

- *Representation*: A label should be a good descriptor of the sound scene or event, of sufficient descriptive power to allow understanding of the sound properties based on the label.
- *Non-ambiguity*: A label should be unequivocal and have a clear one-to-one correspondence to the type of sound to which it applies.

6.2.2.1 Sound Scene Labels

Sound scene labels usually consist of a very short textual description of the scene that provides a meaningful clue for identification: e.g., *park*, *office*, *meeting*, etc. To enable consistent manual annotation, each label needs to be connected to a clearly defined situation, including a specification of the level of detail for the description—to fulfill the necessary representation requirement. A general sound scene may be *home*, to mark scenes happening in the home, but in some cases one may want to differentiate between scenes happening in the *kitchen*, *living room*, etc., therefore using a finer level of detail in drawing distinctions between scenes. After a certain level, it becomes hard to distinguish separate categories, increasing the risk of confusion among human labelers.

Ambiguity in sound scene labels can arise from interpretation of the scene or the label due to personal life experience—the level of traffic that warrants the labeling of a scene as *busy street* is likely to be different for someone living in a densely populated city compared to someone living in a small town, while *train station* can be indoor or outdoor. For non-ambiguity, additional descriptive information of the label may be important to both annotator and end user of the system.

A sound scene label may refer to an entire recording. In this case the main characteristics of the scene around the recording device do not change during the recording, while the recording device can be stationary or moving within the same acoustic scene. Such data is meant for typical *classification* tasks, with the expected system output being a single label that characterizes the entire test example. There are also cases where *segmentation* of an audio recording into different scenes is required—for example, when dealing with long recordings in life-logging applications, in which the scene can change multiple times according to movements of the user carrying the recording device. In this case the annotated scene labels refer to non-overlapping segments of a single recording. It follows that in related applications the expected system output is a segmentation providing the scene change points, and multiple labels, each associated to a segment of the test example. Different granularities of sound scene annotation are illustrated in Fig. 6.1.

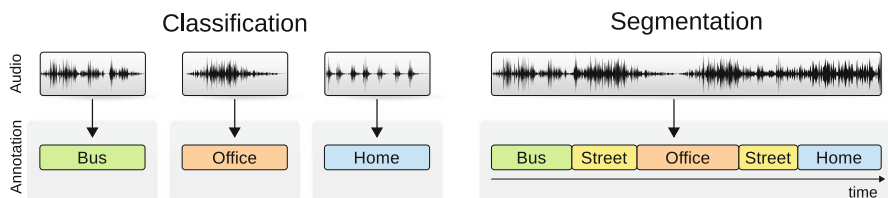


Fig. 6.1 Annotation of sound scenes for classification and segmentation

6.2.2.2 Sound Event Labels

Sound event labels usually consist of a very short and concise description of the sound as perceived by humans. The interpretation of the sound in terms of its source is related to so-called *everyday listening* (introduced in Chap. 3), while the interpretation in terms of its acoustic qualities is related to *musical listening* [18]. According to Gaver [18], sounds are caused by the interaction of materials and convey information about those materials and interactions; at the most basic level sounds can be categorized as involving solids, liquids, or gases, with a few basic-level events (e.g., impacts, pouring, explosions). When asked to describe a sound, people describe it most often in terms of its source, meaning the object or being that produces the sound, or the action that causes the sound, with a more complete description consisting of an object-action pair [5]. Sound event labels characterize the content of a sound scene in terms of individual sounds, for instance, with sound events such as *people talking*, *children shouting*, and *dog barking* being possible elements of a *living room* or *home* sound scene.

Since it depends on personal life experience and perception, the labeling of sound events is highly subjective [22]. The most common everyday sounds will likely be similarly described by people of similar cultural background, but differences may still appear, for example, from using synonyms (*car* vs. *automobile*) or using related words that offer a correct description at a different level of detail (*car* vs. *vehicle*) [21, 29]. Increased detail in descriptions provides specificity, whereas a more general term is more ambiguous. The choice of terms must also fulfill the representation requirement—therefore non-informative textual descriptions like *car noise* are best avoided in favor of labels allowing interpretation of the sound [21], e.g., *car engine running*, *car passing by*, or *car horn*.

Similar to the case of sound scene labels, a sound event label may be attached to an entire recording, indicating the presence of the described sound in the audio without specifying the actual temporal location of the sound event. In this case, the label is referred to as a *weak label* or a *tag*. A single audio example may have multiple associated tags [16]. In its most informative form, a sound event label has associated temporal boundaries to mark the sound event’s onset and offset, making it a *strong label*. In this case, an event label characterizes a certain segment of the audio recording which represents a *sound event instance*, with multiple instances of the same category being possible in the same recording. In real-life situations

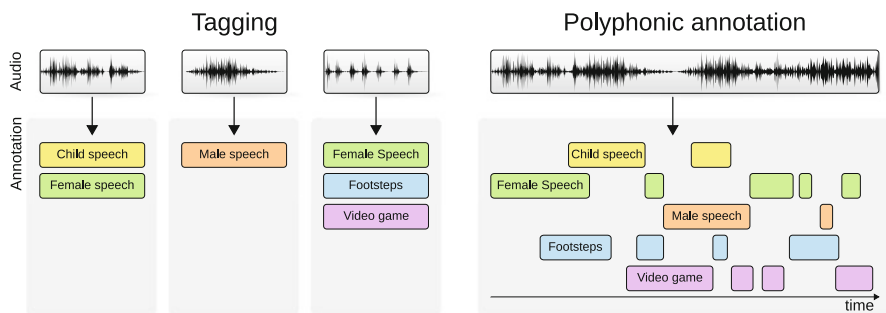


Fig. 6.2 Tagging and polyphonic annotation of sound events

it is very common for multiple sounds to overlap. If multiple overlapping sounds are annotated, the resulting annotation is a *polyphonic annotation* [32], which is practically the most complex form of annotation of environmental audio. Figure 6.2 illustrates these types of annotation for sound events.

6.3 Obtaining Reference Annotations

To allow supervised learning and evaluation, the audio must have corresponding reference annotations. These annotations can be produced manually or in various semi-automatic ways, with the quality and level of detail available in the obtained annotation often depending on the procedure used. Manual annotation involves human annotators that will produce a mapping of the audio content into textual labels. Manually annotating sound *scene* audio material is relatively fast, while for sound *events* the process is much slower, with annotation using weak labels being much faster than with strong labels. Manual annotation is prone to subjectivity arising from the selection of words for labels and placing of temporal boundaries.

Automatic methods for creating annotations may take advantage of the specific content of the audio, for example, using endpoint detection to find segments of interest or using a pre-trained classifier for assigning labels to segments of audio. Automatic methods are prone to algorithm errors; therefore, human verification of the annotations is necessary to ensure that the outcome has sufficient quality. Endpoint detection has been used, for example, to annotate the ITC-irst data [57], with subsequent verification of the sound event boundaries. In the case of synthetically created audio mixtures (discussed below), annotations can be automatically produced at the same time as the audio mixtures.

Another method to obtain annotations is crowdsourcing. Crowdsourcing annotations is possible for certain types of data and can make use of existing tools such as Mechanical Turk,¹ making it a convenient way to obtain labels for

¹Amazon Mechanical Turk, <https://www.mturk.com>.

categorization tasks, where the annotator must only provide or assign a label for a given segment. With crowdsourcing it is possible to obtain many judgments in a short time and to produce high-quality annotations using inter-annotator agreement measures. Crowdsourcing is also appropriate for the verification of weakly labeled data available from sources like Freesound or YouTube, but segmentation, with or without labeling, is less easily achieved via crowdsourcing.

6.3.1 Designing Manual Annotation Tasks

Designing the manual annotation procedure is a difficult part of the data collection process. Ideally, the annotation procedure should be broken down into simple *unit tasks* that can be performed quickly, without requiring much effort from the annotator. A unit task is, for example, labeling of a single segment of audio, or selecting the category from a list of available labels that was decided in advance. When the audio is presegmented, the annotator is free of the burden of positioning event or scene boundaries and needs only concentrate on labeling the presented audio segment. When the categories to be annotated are selected in advance, the annotator is free of the burden of selecting representative and non-ambiguous labels, and only needs to identify and locate the required categories, ignoring other content. These are good methods for simplifying the annotation process by giving a clear task to the annotator, but are not always suitable choices in data collection, as will be detailed below.

6.3.1.1 Annotation with Preselected Labels

This approach frees the annotator from choosing terms for the labels and is applicable to most cases dealing with a specific detection/classification problem. In this case, the labels are selected such that they match the classes defined by the target application, and the annotator only needs to know what kind of content is relevant to each of the provided labels.

Preselected labels will significantly speed up the annotation process for sound scenes, as long as each label has a precise definition to clearly mark the main characteristics on which identification is based. Studies show that human subjects have a very high sound event identification accuracy in a closed-set response format [51]; therefore, this method should result in high-quality category annotations. The annotator can be provided with additional information (e.g., general recording conditions, location, geographical information or physical characteristics) to help in resolving situations that cannot be solved using only the acoustic characteristics. Differentiation between acoustically similar scenes may then be based on the additional information.

When annotating sound events, a predefined set of labels will help annotators use consistent labels. This results in well-defined categories for the research task, but

does not allow any flexibility in differentiating between acoustically or conceptually similar sounds and may cause difficulties when target categories are very similar, for example, verbs expressing objects being hit: *bang*, *clack*, *clatter*, *clink*. The greater the number of classes, the more difficult it is to differentiate between the relevant sounds, which increases the importance of training annotators about what each label means.

Annotation using predefined labels requires careful consideration of the labels and their definitions. While labels can be based on clearly defined characteristics, the definitions must solve ambiguities: for example, when annotating an outdoor cafe scene with no *outdoor cafe* label available in the provided list of labels, is it more appropriate to select *cafe* or *street*? In the case of sound events, it is recommended to select labels indicating clear sound sources or actions such as *air conditioning* or *door slam*, as sources are easy for the annotators to recognize and label, compared to more abstract concepts like *broadband noise* or *percussive sound* which do not facilitate fast identification and interpretation of the sound.

6.3.1.2 Annotation of Presegmented Audio

Another option to speed up the annotation process is to segment the audio in advance—leaving the annotator only the labeling of the segments. Combining this with predefined labels results in a greatly simplified annotator task. The simplest way is to segment the audio into fixed-length segments, without taking into account natural boundaries of its content. Annotation of these segments will provide a coarse time resolution of annotated content [16, 19], which may be sufficient for applications that only require tagging. Such a method is, however, unsuitable for obtaining a detailed annotation that requires segmentation of content into acoustically or conceptually meaningful units such as different scenes or individual sound events.

To obtain such meaningful units through segmentation, one option is to use human annotators to first perform segmentation without labeling, and provide the labeling task later based on the segmentation task's results. Automatic segmentation techniques may also be applicable, but their performance needs to be assessed before trusting them for the annotation task. In general, segmentation techniques are fairly successful for scene segmentation and for sequences of non-overlapping sound events with mostly silent background, but they are likely to underperform for highly polyphonic or noisy mixtures.

6.3.1.3 Free Segmentation and Labeling

Completely unrestricted annotation involves segmentation and labeling performed by the annotator at the same time—allowing free choice of labels and selection of the relevant segment boundaries. Both aspects pose difficulties and introduce subjectivity into the annotations. In this annotation procedure, the level of detail for

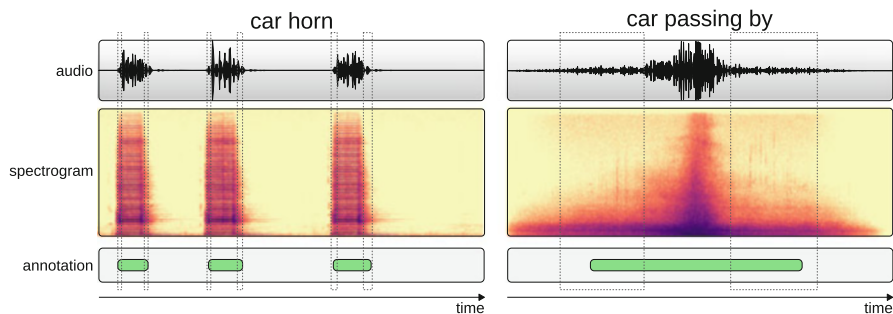


Fig. 6.3 Annotating onset and offset of different sounds: boundaries of the sound event are not always obvious

the label is selected by the annotator and cannot be controlled [21]. The resulting variability in labels may be undesired, but a dataset that is freely annotated is typically not targeted to any specific classification or detection task, allowing full flexibility in its future use and development. Such general data can be adapted to specific applications by narrowing it down to target classes directly based on the available labels, or by postprocessing to map available labels into more uniform annotations. Because there is no restriction on the annotation categories, the annotation process may result in diverse and unbalanced classes, some of which may be arbitrarily rare. A level of control can be set by imposing rules on the labels, such as the use of noun/verb pairs [5], and by devising methods for refining the resulting annotations.

Segmentation subjectivity is mostly related to positioning of onsets and offsets, which may prove difficult for some categories. Figure 6.3 illustrates two sounds: *car horn* and *car passing by*. It is clear that for the *car passing by* sound it is much more difficult to pinpoint an exact onset or offset. The resulting subjectivity in event boundaries could be alleviated by combining multiple annotators' opinions [26] or by using appropriately designed metrics when measuring the boundary detection performance. For example, in speaker diarization the possible subjectivity of annotation at speaker change points is counteracted by allowing a tolerance of, for example, ± 250 ms at the boundaries when evaluating performance [3]. A study on measure annotation for western music indicates a similar tolerance is necessary for this task [55], but up until now there are no systematic studies of this dimension for annotations of sound events.

Performing segmentation and annotation at the same time is quite demanding, yet reduced annotator effort is the key to success in obtaining reliable annotations. Simplification of either aspect (labeling or segmentation) will discard some level of detail that could be obtained, either descriptive or temporal. On one hand, it is important to find solutions that simplify the annotation process, but on the other hand, oversimplification will limit the generality and reusability of the resulting dataset. There will always be a trade-off between the speed and amount of data that can be collected and the quality and detail of annotations; this deserves careful consideration when making a data collection plan.

6.3.2 *Inter-Annotator Agreement and Data Reliability*

Ideally the annotation offers a true representation of the audio content and can be considered as ground truth. In reality, this is not the case, and in the best scenario the annotation is to be trusted as a suitable *gold standard* reference in evaluation. For this reason, testing the reliability of the annotation process should be a common requirement. Reliability measures reflect the trustworthiness of the annotations and help refine the annotation process to produce consistently good annotations. Up until now, work in environmental sound detection and classification has not considered the subject of data reliability, but in other fields, such as computational linguistics, reliability testing is more common.

There are two aspects to be considered: reliability of the annotator and reliability of the produced annotations. On one hand, when the annotation process is well understood by the human annotators, the same annotator should perform consistently and identify the target categories with equal confidence. On the other hand, a good annotation procedure should be independent of the annotator, such that different annotators should identify the target categories in the same way. Measuring inter-annotator variability can offer corrections for improving both the annotation procedure and the quality of annotations by identifying categories on which disagreement is higher and concentrating on better defining the corresponding labels. The previously discussed annotation procedures are based on different rules, and agreement needs to be measured differently for segmentation and for categorization (labeling of fixed segments). Different measures for agreement exist, but there is no universal procedure for how to measure it and how to interpret the obtained values.

In simple categorization problems, in which annotators only have to choose a category for already identified units, inter-annotator agreement can be measured using percentage agreement or specific coefficients such as Cohen's kappa [9], Scott's pi [48] and its generalization for more than two annotators, Fleiss's kappa [14], as well as Cronbach's alpha [10]. Each of these measures attracts some criticism, related to accounting or not for chance agreement, systematic biases introduced by the number of annotators or their use of categories, or the minimum value acceptable for concluding reliability. Nevertheless, in computational linguistics these coefficients are well known and have been widely used since 1950.

An example of an inter-annotator agreement study for environmental sound labels is presented in [16]. Audio segments with a length of 4 s were tagged by three annotators using seven predefined labels. The inter-annotator agreement was measured using Jaccard index which measures similarity between two sets. The study showed significant differences in annotator opinion for the seven available labels. Samples with very low agreement score were eliminated for subsequent use, but many others with relatively low score were kept in the dataset, without providing any guidance for dealing with such cases.

Measuring reliability for segmentation—marking boundaries of units—or simultaneous segmentation and labeling has been mostly neglected, being treated instead

as categorization of fixed length units, thus artificially reducing the measuring of reliability to the same known coefficients. The lack of a methodology to assess annotator reliability and inter-annotator agreement for more complex styles of annotation, including the polyphonic case, is a hurdle that remains to be overcome for creating reliable annotations.

6.4 Datasets for Environmental Sound Classification and Detection

For developing systems that can be successfully applied in real-life applications, the data used in development and testing should be either naturally recorded in similar conditions, or created artificially with sufficient realism to achieve the same result. Training and developing with audio data that resembles the application ensures that the trained models are suitable and the system performs at its best. Specific applications that target certain sound categories may use examples of the sounds recorded in isolation, but often real-life applications will require methods to overcome the presence of concurrent sounds.

The data collection procedure must be defined at the beginning of the process. It may be useful to perform a few test recordings and annotations or test the data selection procedure if the workflow is unclear, or simply to verify all the steps involved. The data collection should be planned according to the desired properties of the outcome, including aspects such as low/high audio quality, single device/multi device, closed/open set, etc. If a target application is already in sight, the audio properties are selected so as to match the expected system use, and the set of classes is selected to match the application needs. In the ideal, using a very general setup to collect a large amount of data will make it possible to subsequently narrow down the dataset for a variety of more specific purposes.

6.4.1 Creating New Datasets

6.4.1.1 Recording New Data

Recording real-world audio is the obvious data collection method for obtaining realistic data. Creating a new dataset by recording new data has the advantage of producing a collection with controlled audio quality and content. When planning the recording process, details such as microphone types, recording devices, audio quality (sampling frequency, bit depth, monaural/binaural) can be decided. Use of the same settings and device(s) throughout the data will result in a uniform quality set.

There are unfortunately some disadvantages to collecting new data. One disadvantage of recording new data is that in order to cover as much acoustic variability

and diversity as possible, recordings must be done in many different conditions. For location-specific modeling, this may mean different weather or human activity conditions, while for more general modeling, it would also require traveling to other locations, adding significant time and effort to the data collection procedure. For the algorithm development stage, however, this is an advantage: A highly diverse dataset gives very good foundation for developing algorithms with good generalization properties. If annotation of the newly recorded data is planned, the annotation process adds to the disadvantages. Depending on the purpose of the data, annotation may be relatively effortless—for example, sound scene annotation for classification—or slow and tedious, as for polyphonic sound event data. As a consequence, this method for creating datasets is more attractive to applications where the recording and possibly annotation effort is offset by the resulting controlled data quality.

Recording new data is popular for sound scene classification, because for this task the single label per recording can be easily provided [12, 33, 53]. In some cases, the annotation process can be embedded in the data recording process, for example, when recording through a mobile application [41]. Collecting and annotating new data for sound event detection is much less common, but it was done, for example, in the DCASE challenges [33, 53].

6.4.1.2 Collecting a Set of Existing Recordings

It is possible to take advantage of already-recorded data by collecting audio from various existing collections—whether freely available or commercial. This approach has the advantage of speed, since it does not involve physically doing any recordings. In this approach there is no direct control on the properties of the audio, and while acoustic diversity can possibly be achieved by collecting a sufficiently large number of examples, the audio quality is something that needs to be reviewed. It is possible to simply disregard audio examples that do not fulfill certain requirements, for example, by imposing a minimum sampling rate or bit rate, but the resulting dataset will likely contain audio recorded with different microphones and devices and will not have uniform quality. Having audio of varying recording condition and quality may be a disadvantage in some situations, but also useful for creating robust algorithms.

This method for creating new datasets is most often used to collect isolated sound examples for sound event classification [46], or for creating artificial mixtures [13] as will be explained next. When creating datasets this way, the usual method is retrieval of examples from the original source collections based on textual queries. Often these queries involve or are the exact target labels. Audio samples are selected from the retrieval output along with their labels or tags. A broader search can be performed by using semantically equivalent or similar terms in the queries, in which case it is necessary to create a mapping between the query terms and the labels assigned in the new dataset to the retrieved examples.

6.4.1.3 Data Simulation

A fast way to produce sound scenes data with reliable annotations is data simulation. Complex audio examples can be created by mixing isolated sound tokens with the desired complexity of overlapping target sounds as well as possible ambient or noise background [28]. This method has the advantage of generating reliable reference annotations during the process of creating the mixture audio, based on the placing of the individual sound event instances. It also offers the possibility of controlling the relative levels of different sound events and background, making it possible to create the same combination of sounds at many different signal-to-noise ratios.

The main disadvantage of this method is that in order to obtain a realistic sound scene it is necessary to consider rules for sound co-occurrence and overlap, thereby creating a kind of “language model” of our everyday environment. While it is possible to generate a wide variety of scenes, this method has the inherent limitation of using a restricted set of source samples that may not fully model the complexity of real world data.

Audio capture in a variety of acoustic environments can be also simulated to introduce diversity into data. A simple way to accomplish *room simulation* is to collect room impulse responses measured from differently sized and shaped rooms, capture the source audio in low reverberant space, then simulate audio capture in different rooms by convolution of the clean audio signal with the available room impulse responses [58, pp. 191]. This process will introduce the reverberation characteristics of the selected room into the acoustic signal. Another possibility to simulate room response is to use image method [2].

6.4.1.4 Typical Pitfalls in Data Collection

To satisfy the need for diversity when recording new audio, the goal is to record multiple instances of the same situation, such as different locations and multiple conditions for each acoustic scene class, e.g., rainy/sunny/windy, winter/summer, crowded/quiet, and multiple examples for each sound event, e.g., footsteps of different people, at different speeds, on different surfaces, etc. For audio scenes, a record of the geographical location should be also kept to be used later in the experiment setup, to avoid training and testing with data from the same location, unless this is specifically desired (location-specific models). If the recorded audio is intended to be cut into smaller pieces and distributed as such (as in the DCASE 2016 Acoustic Scene Classification task [33], as well as the Rouen set [8]), information that identifies which pieces originate from the same underlying recording is necessary for the same reason. This can pose a problem when building datasets from existing audio recordings if there is no such information on common origin available for the collected files.

When creating synthetic audio, a similar issue arises in order to avoid using mixtures containing the same instances in train and test sets. When creating artificial mixtures, another challenge is to create data such that the systems using it will be able to generalize, which means that a large number of source examples with high diversity are necessary.

Perhaps the biggest pitfall of evaluation datasets concerns their size. Typically, the effort involved in designing the task is much less than the labor required to collect and then annotate the data, so the overall cost is roughly linear in the size of the dataset. As these costs can be significant (months of annotator effort, for example), there is a natural tendency to create the smallest acceptable dataset. However, a small dataset provides few examples (particularly for rarer classes), meaning that estimates of performance are correspondingly “noisier” (exhibit high variance). “Noisy” performance metrics can be corrosive and misleading, resulting in spurious conclusions about the merits of different approaches whose relative performance may be entirely due to chance. A rigorous research process includes estimating the confidence intervals of any measures, and/or the statistical significance of conclusions. However, these measures can be difficult to calculate for some metrics and are too frequently overlooked [56]. Researchers are advised to run evaluations with many minor variations of system parameters or random initializations to acquire at least some sense of the scale of variation of performance metrics that can occur at random.

6.4.2 Available Datasets

As explained above, benchmark datasets are important in research for comparing algorithms and reproducing results in various conditions. Currently available datasets include all previously mentioned dataset creation methods: data specifically recorded for a certain task, data retrieved from existing free-form collections, and data artificially created to model different degrees of complexity in everyday environments.

Free-form collections are a valuable resource for creating specific datasets. Freesound² is an example of such a collection, its most important asset being that it is freely available. Of the disadvantages, probably the most apparent is the unstructured annotation of the data, but many studies have performed a careful selection or filtering of the labels when creating the datasets. Freesound is the source for the ESC datasets [38], NYU UrbanSound [46], and Freefield [52]. Other audio data sources are the commercially available audio samples from BBC, Stockmusic, and others.

A list of freely available datasets for sound classification and tagging and sound event detection is presented in Table 6.1. The list is not meant to be comprehensive (particularly since many new datasets are appearing at this time), but it provides examples that illustrate disparities between different datasets.

Among the sound scene datasets in Table 6.1, Dares G1 [21] has been recorded for general environmental sound research, but due to free annotations resulting in over 700 sound event classes (distributed among only around 3000 annotations),

²www.freesound.org.

Table 6.1 Datasets for sound classification, tagging, and sound event detection. Datasets for sound scene classification are newly recorded (rec), while datasets for sound event classification are often collected (col) from available repositories. Most sound event detection datasets are newly recorded or produced synthetically (syn)

	Dataset name	Type	Classes	Examples	Size (min)	Usage, publications
Sound scenes	Dares G1	rec	28	123	123	[21, 31]
	DCASE 2013 Scenes	rec	10	100	50	[53]
	LITIS Rouen	rec	19	3026	1513	[8, 40]
	TUT Sound Scenes 2016	rec	15	1170	585	DCASE 2016, [33]
	YouTube-8M	col	4716	>7M	>27M	[1]
Environmental sounds	ESC-10	col	10	400	33	[25, 37]
	ESC-50	col	50	2000	166	[37, 38]
	NYU Urban Sound8K	col	10	8732	525	[46]
	CHIME-Home	rec	7	6137	409	DCASE 2016, [16]
	Freefield1010	col	7	7690	1282	[52]
	CICESE Sound Events	col	20	1367	92	[6]
	AudioSet	col	632	>2M	>340k	[19]
Sound events	Dares G1	rec	761	3214	123	[21, 31]
	DCASE 2013 Office Live	rec	16	320	19	DCASE 2013, [53]
	DCASE 2013 Office Synthetic	syn	16	320	19	DCASE 2013, [53]
	TUT Sound Events 2016	rec	18	954	78	DCASE 2016, [33]
	TUT Sound Events 2017	rec	6	729	92	DCASE 2017
	NYU Urban Sound	col	10	3075	1620	[43, 44, 46]
	TU Dortmund Multichannel	rec	15	1170	585	[27]

it is rather difficult to use for sound classification. DCASE 2013 [53] datasets are balanced, but they are very small. LITIS Rouen [40] and TUT Sound Scenes [33] are larger sets but formed of 30 s segments cut from longer recordings, and while providing a recommended cross-validation setup, LITIS Rouen does not include complete information linking the segments to the original recordings, leading to difficulties in the experimental setup. The YouTube-8M collection of audio and video features includes a very large number of classes, but these are not specifically audio, nor even specific scenes, being categories assigned to the videos on YouTube such as “Animation” or “Cooking” [1]. The datasets of environmental sounds are generally larger than the others, most being created by collecting audio from Freesound or YouTube [19]. These sites also provide rich sources for unsupervised learning approaches [4].

Datasets for sound event detection are the most varied, including newly recorded and annotated data, as well as synthetically generated mixtures and data collected from Freesound. DCASE 2013 and 2016 synthetic datasets contain overlapping sound events and have polyphonic annotation, DCASE 2013 Office Live contains sequences of events separated by silence, NYU Urban Sound contains a single event instance, while TUT Sound Events 2016 and 2017 contain overlapping sound events recorded in everyday situations and with polyphonic annotation.

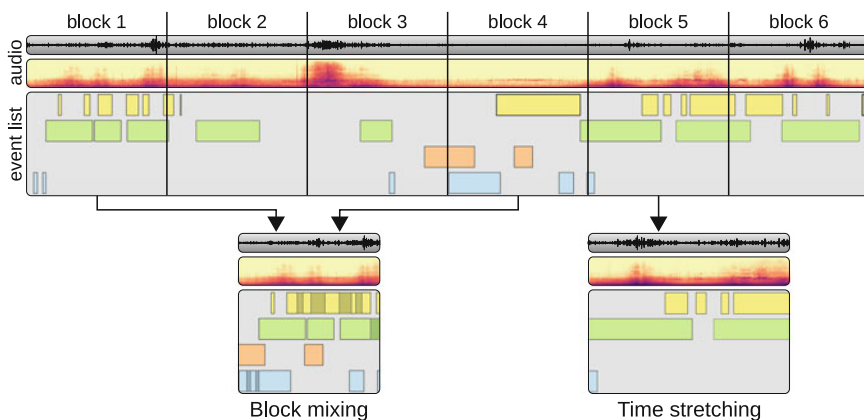


Fig. 6.4 Data augmentation by block mixing and time stretching

6.4.3 Data Augmentation

Data augmentation refers to methods for increasing the amount of development data available without additional recordings. With only a small amount of data, systems often end up overfitting the training data and performing poorly on unseen data. Thus, it is desirable to artificially increase the amount and diversity of data used in training.

Approaches for modifying existing data include straightforward methods like time stretching, pitch shifting or dynamic range compression [45], or convolution with various impulse responses to simulate different microphones and different room response. More complicated modifications are sub-frame time shifting [35], frequency boosting/attenuation, mixing of sounds from same class, or random block mixing [54]. Other modifications can include mixing of available audio with external data to add background noise with various SNRs, simulating different noise conditions.

Block mixing and time stretching procedures are illustrated in Fig. 6.4. Block mixing is based on the superposition of the individual waveforms of two or more sound sources active at the same time. Based on this, new signals can be created by summing sections of the original recordings. For the resulting data, labels are created as the framewise union of the set of labels of the original recordings. Under the serviceable assumption of linearity for the magnitude spectra, the spectrogram of the new data can be obtained by direct summing of the spectrograms of the two sections to be combined, shortening the feature extraction process by avoiding calculation of the spectrogram of the mixture. This can then be further processed to obtain the desired feature representation.

6.5 Evaluation

Evaluation is usually framed as estimating the performance of a system under test when confronted with new data. For an objective evaluation, the system is fed previously unseen data for which reference annotations are available. The system output is then compared to the reference to calculate measures of its performance. What performance means and how it should be measured may vary depending on the specifications and requirements of the developed system: We can measure accuracy to reflect how often the system correctly classifies or detects a sound, or we can measure error rates to reflect how often the system makes mistakes. By using the same data and the same methodology to evaluate different systems (perhaps in different places and/or at different times), a fair and direct comparison can be made of systems' capabilities.

Subjective evaluation of systems is also possible and relies on human ratings of the system performance. Subjective evaluation often involves listening experiments, combined with agreement or satisfaction ratings of the system outputs. Subjective evaluation is useful when there is no available reference annotation for the test data, or the system clearly targets customer satisfaction. Below, we limit our discussion to objective evaluation, since measuring user experience through qualitative methods is application-specific and out of the scope of this book, being more common in usability engineering and human-computer interaction.

6.5.1 Evaluation Setup

During system development, iterative training and testing of the system is necessary for tuning its parameters. For this purpose, the labeled data available for development is split into disjoint training and testing sets. However, labeled data is often in short supply, and it is difficult to choose between using more data for training (leading to better-performing systems) or for testing (giving more precise and reliable estimates of system performance). For an efficient use of all the available data, system development can use cross-validation folds [11, p. 483] to artificially create multiple training/testing splits using the same data; overall performance is then the average of performance on each split. Cross-validation folds also help avoid overfitting and supports generalization properties of the system, so that when the system is used on new data it is expected to have similar performance as seen with the data used for development.

If available, a separate set of examples with reference annotation can be used to evaluate the generalization properties of the fully tuned system—we refer to this set as *evaluation set*, and use it to evaluate how the system would perform on new data. Figure 6.5 illustrates an example partitioning of a dataset into development data and evaluation data, with further partitioning of the development data into training and testing subsets in five folds. The split here is done so that all data is tested at most one

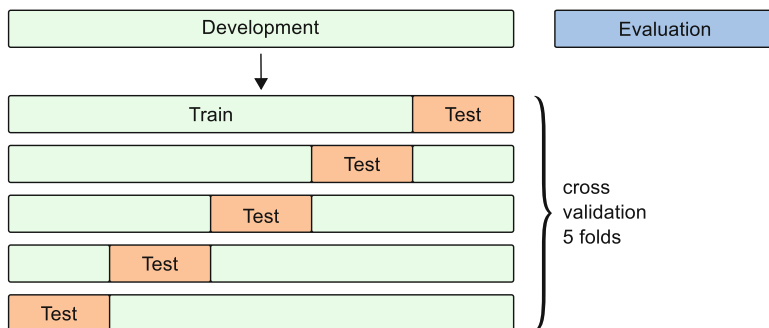


Fig. 6.5 Splitting a dataset into development and evaluation data, with five folds for cross-validation during system development

time, but it is also possible to create the train/test subsets by randomly selecting data into each subset. There are, however, some details to take into account, depending on the way the original dataset was created.

When splitting a dataset into train and test subsets, metadata available for the audio examples is useful for ensuring that data coming from same location (segments of the same original long recording), or data with same content (synthetic data in which same instance has been used) is never split across train and test sets. This is necessary in order to ensure that the system learns the general properties of the data, rather than the details of specific instances.

Stratification of the data is recommended, if possible. This means splitting the data into folds in a supervised manner, such that each fold is representative for the data. The aim of stratification is to ensure that the train and test sets used in development contain balanced data, with all classes present in all folds, with similar amounts of data for each class [50]. Unfortunately, this is not always feasible, especially with multilabel classification or polyphonic sound event detection. Most often, the available data is not perfectly balanced, and in this case the train/test splits should be constructed at least to ensure that there are no classes being tested that were not present in the corresponding training data.

Repeated evaluation of the system is often necessary during its development, which involves calculating the metrics of choice after testing the system. With unbalanced data, different folds will have different amounts of data for classes, or even classes completely missing from the test set of a given fold. In such cases, calculating the overall performance as an average of the fold-wise performance will depend on how data is distributed within the folds and will differ for any different split. This can be avoided by treating the folds as a single experiment and evaluating system performance only after performing the complete cross-validation, thus ensuring that all classes are fully represented in the test set [15].

6.5.2 Evaluation Measures

Evaluation is done by comparing the system output with the reference annotations available for the test data. Metrics used in detection and classification of sound scenes and sound events include accuracy, precision, recall, F-score, receiver operating characteristic (ROC) curve, area under the curve (AUC), acoustic event error rate (AEER), or simply error rate (ER). There is no consensus over which metric is universally good for measuring performance of sound event detection, as they each reflect different perspectives on the ability of the system.

6.5.2.1 Intermediate Statistics

Many performance measures rely on *trials*, the notion of atomic opportunities for the system to be correct or make a mistake, and the metrics are calculated based on counts of the correct predictions and different types of errors made by the system. These counts are referred to as *intermediate statistics* and are defined depending on the evaluation procedure. Given class c they are defined as follows:

- *True positive*: A correct prediction, meaning that the system output and the reference both indicate class c present or active.
- *True negative*: The system output and the reference both indicate class c not present or inactive.
- *False positive* or *insertion*: The system output indicates class c present or active, while the reference indicates class c not present or inactive.
- *False negative* or *deletion*: The system output indicates class c is not present or inactive, while the reference indicates class c present or active.

A false positive can appear at the same time as a false negative, when the system output indicates class c while the reference indicates class g . In this situation, some metrics consider that the system makes a single error—a *substitution*—instead of two separate errors. More details on how substitutions are defined for specific cases are presented with the description of the metrics that use them.

The intermediate statistics are calculated based on the trial-wise comparison between the system output and the available reference for the test data [47]. Individual trials can be formed by specific items identified in the annotations, or simply from fixed length intervals. The fixed length intervals can be any small temporal unit, such as a frame of 20–100 ms, similar to the typical audio analysis frame [39, 53], or a longer interval such as 1 s [24]. In item-level comparisons the item can be an entire audio clip from start to end, or a sound event instance. Corresponding metrics are referred to as *segment-based* metrics and *item-based* or *event-based* [32] metrics in the case of sound events.

Acoustic scene classification is usually a single-label multiclass problem, and the resulting intermediate metrics reflect whether the single true class is correctly recognized for each example. In this task there is no role for substitutions, and each

erroneous output is counted; there is no distinction between false positives and false negatives. In scene segmentation the intermediate statistics can be counted in short time intervals or based on segmentation boundaries.

In sound event detection, the choice of measurement determines the interpretation of the result: With a segment-based metric, the performance shows how well the system correctly detects the temporal regions where a sound event is active; with an event-based metric, the performance shows how well the system is able to detect event instances with correct onset and offset. If we consider the scenario of polyphonic sound event detection, the segment-based metric essentially splits the duration of the test audio into fixed length segments that have multiple associated labels, reflecting the sound events active anywhere in the given segment. In this respect, evaluation verifies if the system output and reference coincide in the assigned labels, and the length of the segment determines the temporal resolution of the evaluation. Event-based metrics compare event instances one to one. Since the time extents of the events detected by the system may not exactly match the ground truth, a common approach is to allow a time misalignment threshold, either as a fixed value (e.g., 100 ms) or as a fixed proportion of the total event duration (e.g., 50%) [53]. Time thresholds can be applied to onset times only, or to both onset and offset times. True negatives (i.e., correctly recording that no event occurred at a given time) do not occur in this kind of evaluation, as no instance-based true negatives can be counted. Figure 6.6 illustrates the comparison between a system output and the reference in segment-based and event-based evaluations, for obtaining the intermediate statistics.

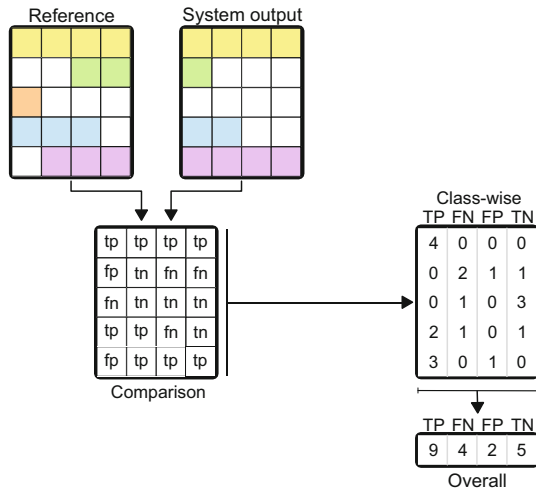
6.5.2.2 Metrics

Measures of performance are calculated based on accumulated values of the intermediate statistics. We denote by TP, TN, FP, and FN the sums of the true positives, true negatives, false positives, and false negatives accumulated throughout the test data. With same convention, we will use S for the total number of substitutions, I for insertions, and D for deletions.

Considering a simple binary classification, where we compare the reference and the system output for class c , we can construct the *contingency table*, or *confusion matrix* presented in Fig. 6.7. Based on the total counts of the intermediate statistics, many different measures can be derived, of which the most commonly used are *recall* (R , also known as *true positive rate* (TPR) or *sensitivity*), *precision* (P), *false positive rate* (FPR), *specificity*, and *accuracy* (ACC). These measures are presented and defined in Fig. 6.7.

When dealing with a multiclass problem, accumulation of intermediate statistics can be performed either globally or separately for each class [49], resulting in overall metrics calculated accordingly as instance-based or class-based. Highly unbalanced classes or individual class performance can result in very different overall performance when calculated with the two methods. In instance-based averaging, also called *micro-averaging*, intermediate statistics are accumulated over the entire data. Overall performance is calculated based on these, resulting in metrics

Segment-based intermediate statistics



Event-based intermediate statistics

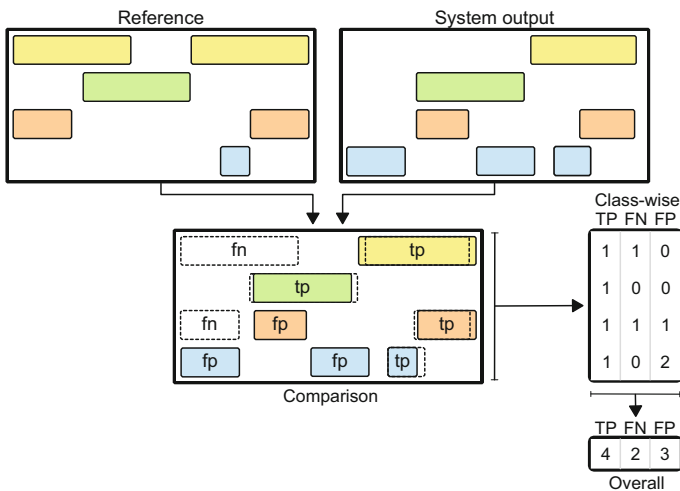


Fig. 6.6 Comparing system output with reference for polyphonic sound event detection in segment-based and event-based evaluation to calculate the intermediate statistics

with values that are most strongly affected by the performance on the most common classes in the considered problem. In class-based averaging (the results of which are also known as *balanced* metrics), also called *macro-averaging*, intermediate statistics are accumulated separately for each category (scene or event class), and used to calculate class-wise metrics. Overall performance is then calculated as the average of class-wise performance, resulting in values that emphasize the system behavior on the smaller classes in the considered problem. A hybrid approach is to

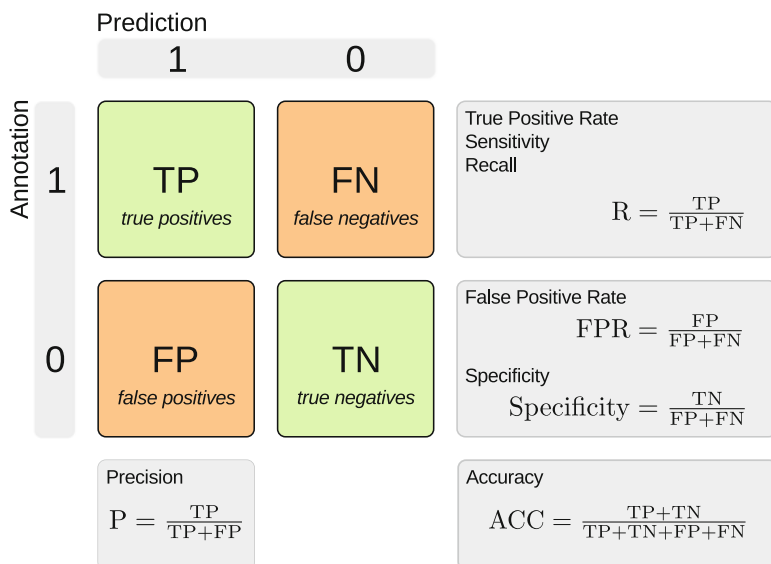


Fig. 6.7 Contingency table and derived measures most commonly used in classification

calculate class-based metrics, then to use nonuniform weighting to combine them to create an average that reflects some relative importance of the different classes that may be different from their frequency of occurrence in the test data.

Accuracy Accuracy measures how often the classifier makes the correct decision, as the ratio of correct system outputs to total number of outputs. As shown in Fig. 6.7, accuracy is calculated as:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (6.1)$$

Accuracy has the advantage of offering a simple measure of the ability of the system to take the correct decision; it is the most-used metric in sound scene and sound event classification, being straightforward to calculate and easy to understand. It has, however, a critical disadvantage of being influenced by the class balance: for rare classes (i.e., where $TP + FN$ is small), a system can have a high proportion of true negatives even if it makes no correct predictions, leading to a paradoxically high accuracy value. Accuracy does not provide any information about the error types (i.e., the balance of FP and FN); yet, in many cases these different types of error have very different implications.

Precision, Recall, and F-Score Precision, recall, and F-score were introduced in [42] in the context of information retrieval, but have found their way into measuring performance in other applications. Precision and recall are also defined in Fig. 6.7:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN} \quad (6.2)$$

and based on them, balanced F-score is calculated as their harmonic mean:

$$F = \frac{2}{1/P + 1/R} = \frac{2PR}{P + R} \quad (6.3)$$

Precision and recall are the preferred metrics in information retrieval, but are used also in classification under the names *positive prediction value* and *sensitivity*. In detection theory terms, recall is equal to *true positive rate*, but precision has no simple equivalent.

F-score has the advantage of being a familiar and well understood metric. Its main drawback is that its value is strongly influenced by the choice of averaging and the data balance between classes: in instance-based averaging the performance on common classes dominates, while in class-based averaging (balanced metrics) it is necessary to at least ensure presence of all classes in all folds in the test data, to avoid cases when recall is undefined (when $TP + FN = 0$); estimates of metrics on classes with very few examples are also intrinsically noisy. Any dataset of real-world recordings will most likely have unbalanced event classes; therefore, the experiment setup must be built with the choice of metric in mind.

Average Precision Because precision and recall rely on hard decisions made for each trial, they typically depend on a threshold applied to some underlying decision variable, such as a distance from a decision boundary or the output of a neural network. Lowering the threshold will increase likelihood of accepting both positive and negative examples, improving recall but in many cases hurting precision. F-measure combines these values at a single threshold in an attempt to balance this tradeoff, but a fuller picture is provided by plotting precision as a function of recall over the full range of possible thresholds—the precision–recall (P-R) curve. Figure 6.8 shows an example of P-R curve for a binary classification problem.

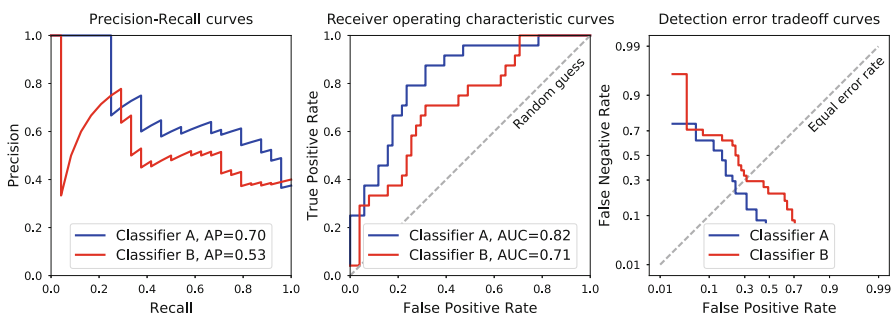


Fig. 6.8 Examples of PR, ROC, and DET curves for evaluating and comparing two classifiers. All three curves show classifier A as superior, as reflected in the AP and AUC values, as well as the equal error rate

While P-R curves carry rich information, they can be difficult to compare, so a single figure of merit summarizing the precision–recall tradeoff is desirable. The information retrieval community most commonly uses *average precision* (AP), which is defined as the precision averaged over all thresholds at which a new positive example is accepted:

$$\text{AP} = \frac{1}{\text{NP}} \sum_{\theta \in \Theta_p} \frac{\text{TP}(\theta)}{\text{TP}(\theta) + \text{FP}(\theta)} \quad (6.4)$$

where $\text{NP} = \text{TP} + \text{FN}$ is the total number of positive examples, Θ_p is the set of NP thresholds at which a new positive example is accepted, and $\text{TP}(\theta)$ and $\text{FP}(\theta)$ are the true and false positive counts, respectively, of the system running with threshold θ . This value is close to the area under the P-R curve; however, it avoids problems that arise from the non-monotonicity of the curve—while recall grows monotonically as the threshold is reduced, precision may go up or down, depending on the balance of positive and negative examples being included under the new threshold. Compared to the single operating point (i.e., threshold) summarized by an F-score, AP reflects performance over the full range of operating points. Because it is based on a larger set of measurements (integrating a curve), it is typically more stable (less noisy) than point measures such as F-score, which is one reason for its popularity. Average precision combined across the categories of a multi-class problem is called *mean average precision*, or mAP.

ROC Curves and AUC The *receiver operating characteristic* (ROC) curve and corresponding *area under the curve* (AUC) are used to examine the performance of a binary classifier over a range of discrimination thresholds. An ROC curve, as illustrated in Fig. 6.8, plots the *true positive rate* (TPR) as a function of the *false positive rate* (FPR), or sensitivity vs. (1–specificity) as the decision threshold is varied. TPR and FPR are calculated as shown in Fig. 6.7:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} = 1 - \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (6.5)$$

AUC summarizes the entire ROC curve into a single number and permits the comparison of classifiers across all operating points, with better classifier having a higher AUC. AUC can equivalently be specified by the more intuitively comprehensible d-prime, defined as the separation between the means of two unit-variance Gaussians whose ROC curve yields the given AUC. A related measure is equal error rate (EER), which is the point on the ROC curve where true positive rate and false positive rate are equal (i.e., the intersection of the ROC curve with the $y = 1 - x$ line); therefore, a better classifier has a smaller EER. Like F-measure, EER is a *point measure* and is typically more variable than AUC, which integrates over a range of operating points. EER has the advantage, however, of being expressed directly in terms of an interpretable value, the classification error rate (of both positive and negative examples).

An alternative to the ROC curve is the DET curve (for Detection Error Tradeoff) [30]. A DET curve, as shown in the third pane of Fig. 6.8, plots *false negative rate* ($\text{FNR} = 1 - \text{TPR}$) as a function of FPR, where both axes are warped by the “probit” function. If the underlying class-conditional score distributions are Gaussian, the resulting plot becomes a straight line whose slope reflects the relative variance of the score distribution for positive and negative classes, and whose intercept with the $y = x$ line indicates overall class separability.

The main disadvantage of ROC curves is that they are only applicable to binary classifiers, and generalizations for multiclass problems are not well defined. The most usual generalization, as with precision/recall measurements, is to treat each class performance separately as a binary classifier output, and calculate the performance as average of the class-wise AUC or EER. This method conceals any variations in performance across classes in the service of producing a single figure of merit to characterize the system.

Error Rate Error rate quantifies errors in the system output with respect to the reference. The precise definition of an error varies to reflect the target of the evaluation; for example, speech recognition uses *word error rate* (WER) which measures the proportion of words erroneously recognized, whereas speaker diarization uses *diarization error rate* (DER) to measure temporal errors as the fraction of time that is not correctly attributed to the appropriate speaker. Error rate has been adapted to sound event detection, but with separate definitions for the intermediate statistics in segment-based and event-based evaluation procedure.

At the segment level, the joint occurrence of a false positive and a false negative is merged to be a substitution (i.e., an event was detected but it was given the wrong identity), without needing to designate which false positive substitutes which false negative. Any remaining false positives in the system output are counted as insertions, or if there are remaining unrecognized reference events, they are counted as deletions. For segment k , this can be expressed mathematically as follows:

$$\begin{aligned} S[k] &= \min(\text{FN}[k], \text{FP}[k]) \\ D[k] &= \max(0, \text{FN}[k] - \text{FP}[k]) \\ I[k] &= \max(0, \text{FP}[k] - \text{FN}[k]) \end{aligned} \quad (6.6)$$

For segment-based evaluation, the number of substitutions, deletions, and insertions are calculated segment by segment and accumulated for all test data.

Event-based intermediate statistics are determined based on the temporal location and label of the detected sound events with respect to the location and label of reference events. A detected event is considered a true positive if a reference event with the same label is present at a temporal location within the allowable misalignment. Usually this allowance is referred to as a *collar* expressed in time units (ms), or replaced by a minimum required event length. Examples of conditions for a true positive could be: the same collar for onset and offset, or a collar only for onset with no offset condition, or a collar for onset and a minimum duration for the event as offset condition. With no consensus on the appropriate values of

the collar or minimum required event length, these parameters of the metric are left to the choice of the developer. Correctly aligned but mislabeled sound events are considered substitutions, while mislabeled or misaligned sound events are false positives (if in the system output) or false negatives (if in the reference). The count of event-based D , I , S is accumulated for all test data.

Error rate is calculated based on the overall D , I , S values as:

$$ER = \frac{D + I + S}{N} \quad (6.7)$$

where N is the total reference “count”; for segment-based ER, N is the sum of active events accumulated segment by segment, while for event-based ER, N is the number of reference events. Because the calculation is made with respect to the number of events in the reference, ER can be larger than 1.0. This can happen even for a system that makes many correct predictions, if it simultaneously makes a large number of false positives (insertions).

As well as characterizing the system performance with a single value, ER has the attraction of paralleling similar error rate measures in other areas. On the other hand, the multiple different definitions (depending on use) can confuse researchers. Note also that a degenerate system emitting no outputs has $ER = 1$ (because it makes N deletion errors); a system that performs some useful classifications while committing sufficient insertion errors to push its $ER > 1$ appears worse than doing nothing by this measure, even though other measures may reflect its actual achievements.

Normalized Decision Cost To control the relative influence of specific errors and correct predictions on the performance measure, it is possible to use a weighting of the intermediate statistics. For example, by assigning weights to TP and TN when calculating accuracy, we obtain *balanced accuracy*:

$$BACC = w \times \frac{TP}{TP + FN} + (1 - w) \times \frac{TN}{TN + FP} \quad (6.8)$$

With equal weights ($w = 0.5$), the balanced accuracy is the arithmetic mean of sensitivity and specificity.

Weighting the intermediate statistics offers the advantage of tuning the chosen metric for the relative significance of error types, but requires some principle for assigning different costs to the different outcomes. The disadvantage of this approach is that it results in many different metrics rather than a universally comparable value and thus complicates system comparison and selection.

Weighted measures also offer an opportunity to soften the problematic time-misalignment collar in event-based metrics. Rather than having a binary decision that an event is correct if it occurs within the collar time of the reference, but incorrect beyond that, a per-event weighting can be applied that ramps up the contribution to the normalized cost from zero to the cost of an insertion plus a deletion as the timing misalignment grows from some minimum value to the maximum collar.

6.6 Advice on Devising Evaluation Protocols

By presenting a crisp target for optimization, evaluation metrics often have a profound influence on the development of a field. If a particular topic or problem is sufficiently interesting to deserve sustained attention to its solution, perhaps from several independent research groups, it is very important to devise and agree shared, repeatable performance measures to avoid publications degenerating into empty argument. But once a well-defined evaluation task is constructed, it can exert an overpowering influence on most or all researchers involved to recast their own work towards optimizing for that evaluation—not least because funders, like DARPA in the original NIST speech recognition evaluations—find direct quantitative metrics very reassuring and thus measurably good performance can become an existential priority. Moreover, the availability of a well-defined evaluation can help attract new researchers to a field, which is very healthy; however, newcomers may naturally assume that the evaluation indicates the only important problems in a field.

For all these reasons, devising evaluation procedures for some field of interest is both an urgent priority that should be addressed sooner rather than later, but also something incredibly delicate that should be considered carefully and not rushed. In the preceding sections, we have examined the different considerations of data and labels, and defined a range of commonly used measures. In conclusion, we offer some summary points of advice for devising evaluations.

Task Choice Choose a task that is as close as possible to something that is useful in itself, rather than some more contrived proxy. Of course, a more realistic task is more likely to share all the properties of real applications (including possibly unrecognized factors in the real-world data). But more importantly, at the end of the day, a specific evaluation task will attract an enormous amount of effort on optimizing for that one task, only some of which may generalize to other tasks. If excessive attention is to be paid to one task, it may as well be something that could actually be useful beyond the evaluation.

Data Amount Go for quantity. Quality is of course important, but there is a strong risk of underestimating the size required to make an evaluation dataset useful. Systems improve with time, and what may have been hundreds or thousands of errors initially may drop to tens or fewer as systems mature. When systems are differing in their results on only a handful of test samples, the value of the task in differentiating performance is lost. Also, tasks can benefit from multiple separate evaluation sets, for instance, to “refresh” a task after participants have begun to overfit a particular set, and if these replacement tasks are drawn from a large pool of data that was all collected in a single effort under the same conditions, evaluation measures will have better continuity.

Statistical Significance Pay attention to statistical significance. Look for a significance test that can be applied to your domain (for example, error counts that are binomially distributed according to a fixed per-trial probability of error), and use it to calculate the theoretical limits of discriminability of your evaluation. In

parallel, make measurements across multiple supposedly equivalent versions of your system and task (varying initialization, splits of data into train/test, small changes in system parameters, etc.) to get an empirical sense of the “error bars” in your metrics. For instance, when using the cross-validation approach illustrated in Fig. 6.5, while averaging the per-fold performance gives a result more stable than that of any single fold, reporting the variance among the individual folds’ contributions to the average can serve as a useful confidence interval.

Baseline System Compare results against a well-established technique, and also against a random or degenerate baseline. If your system is worse than “doing nothing” (e.g., if your error rate exceeds the 100% deletion of simply reporting nothing), then either reconsider your approach (!), or choose a different metric that better reflects whatever useful thing you believe your system is doing. Be very careful when comparing metrics across different datasets, even if they are related. For instance, precision (and hence average precision) is strongly influenced by the underlying prior probability of the target class. Doubling the size of an evaluation set by adding more, consistently distributed negative examples while keeping the same set of positives ought not to change ROC measures like AUC or F-score, but it will halve precision measures, all else being equal.

Metric Choice Choose one or more metrics that emphasize what you care about. Sometimes it is only after comparing the results of different metrics with more qualitative impressions of system performance that you can gain a full understanding of which metrics best track your intentions. Sometimes the best metric may be very specific, such as true positive rate at a particular value of false alarm rate, set according to research into what customers will accept. It is a good idea to start with a diverse set including more metrics than you need; practical experience will then show you which ones are redundant, which are noisy, and which give the best correlation with useful systems.

Error Analysis Recognize that errors are not all created equal: False alarms (insertions) and false rejects (misses) are almost never equivalently undesirable, and normally some classes (and confusions) are much more important than others. Measures like NDC can be constructed to give different costs for each of these outcomes; while the ideal weights may be difficult to establish, even guesses are likely better than implicitly assuming that every error counts equally.

In conclusion, there is no simple formula for devising an evaluation procedure, and there is rarely a way to construct it without investing substantial resources. Ultimately, the real value of evaluations may take years to appear, so the process is inevitably one of trial and error. We hope this chapter has at least illuminated some of the choices and considerations in this critically important task.

References

1. Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., Vijayanarasimhan, S.: Youtube-8m: a large-scale video classification benchmark (2016). arXiv preprint arXiv:1609.08675. <http://research.google.com/youtube8m/>
2. Allen, J.B., Berkley, D.A.: Image method for efficiently simulating small-room acoustics. *J. Acoust. Soc. Am.* **65**(4), 943–950 (1979). doi:10.1121/1.382599
3. Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., Vinyals, O.: Speaker diarization: a review of recent research. *IEEE Trans. Audio Speech Lang. Process.* **20**(2), 356–370 (2012)
4. Aytaç, Y., Vondrick, C., Torralba, A.: Soundnet: learning sound representations from unlabeled video. In: *Advances in Neural Information Processing Systems*, pp. 892–900 (2016). <http://projects.csail.mit.edu/soundnet/>
5. Ballas, J.A.: Common factors in the identification of an assortment of brief everyday sounds. *J. Exp. Psychol. Hum. Percept. Perform.* **19**(2), 250 (1993)
6. Beltran, J., Chavez, E., Favela, J.: Scalable identification of mixed environmental sounds, recorded from heterogeneous sources. *Pattern Recogn. Lett.* **68**, 153–160 (2015)
7. Bertin-Mahieux, T., Ellis, D.P.W., Whitman, B., Lamere, P.: The million song dataset. In: *12th International Society for Music Information Retrieval Conference (ISMIR)* (2011)
8. Bisot, V., Essid, S., Richard, G.: Hog and subband power distribution image features for acoustic scene classification. In: *2015 European Signal Processing Conference (EUSIPCO)*, Nice, pp. 724–728 (2015)
9. Cohen, J.: A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.* **20**(1), 37–46 (1960)
10. Cronbach, L.J.: Coefficient alpha and the internal structure of tests. *Psychometrika* **16**(3), 297–334 (1951)
11. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. Wiley, New York (2000)
12. Eronen, A.J., Peltonen, V.T., Tuomi, J.T., Klapuri, A.P., Fagerlund, S., Sorsa, T., Lorho, G., Huopaniemi, J.: Audio-based context recognition. *IEEE Trans. Audio Speech Lang. Process.* **14**(1), 321–329 (2006)
13. Finney, N., Janer, J.: Soundscape generation for virtual environments using community-provided audio databases. In: *W3C Workshop: Augmented Reality on the Web*. W3C, Barcelona (2010)
14. Fleiss, J.L.: Measuring nominal scale agreement among many raters. *Psychol. Bull.* **76**(5), 378–382 (1971)
15. Forman, G., Scholz, M.: Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *SIGKDD Explor. Newsl.* **12**(1), 49–57 (2010)
16. Foster, P., Sigtia, S., Krstulovic, S., Barker, J., Plu: Chime-home: a dataset for sound source recognition in a domestic environment. In: *Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (2015)
17. Garofolo, J.S., Lamel, L.F., Fisher, W.M., Fiscus, J.G., Pallett, D.S., Dahlgren, N.L.: *DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM*. National Bureau of Standards (1993)
18. Gaver, W.W.: How do we hear in the world? Explorations in ecological acoustics. *Ecol. Psychol.* **5**(4), 285–313 (1993)
19. Gemmeke, J.F., Ellis, D.P.W., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: an ontology and human-labeled dataset for audio events. In: *Proceedings of IEEE ICASSP* (2017)
20. Goto, M., Hashiguchi, H., Nishimura, T., Oka, R.: RWC music database: popular, classical, and jazz music databases. In: *Proceedings of 3rd International Conference on Music Information Retrieval*, pp. 287–288 (2002)
21. Grootel, M., Andringa, T., Krijnders, J.: DARES-G1: database of annotated real-world everyday sounds. In: *Proceedings of the NAG/DAGA Meeting* (2009)

22. Gygi, B., Shafiro, V.: Environmental sound research as it stands today. *Proc. Meetings Acoust.* **1**(1) (2007)
23. He, H., Garcia, E.A.: Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **21**(9), 1263–1284 (2009)
24. Heittola, T., Mesaros, A., Eronen, A., Virtanen, T.: Context-dependent sound event detection. *EURASIP J. Audio Speech Music Process.* **2013**, 1 (2013)
25. Hertel, L., Phan, H., Mertins, A.: Comparing time and frequency domain for audio event recognition using deep learning. In: *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN 2016)* (2016)
26. Krijnders, J.D., Andringa, T.C.: Differences between annotating a soundscape live and annotating behind a screen. In: *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, pp. 6125–6130. Institute of Noise Control Engineering, Ames, IA (2010)
27. Kürby, J., Grzeszick, R., Plinge, A., Fink, G.A.: Bag-of-features acoustic event detection for sensor networks. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, pp. 55–59 (2016)
28. Lafay, G., Lagrange, M., Rossignol, M., Benetos, E., Roebel, A.: A morphological model for simulating acoustic scenes and its application to sound event detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(10), 1854–1864 (2016)
29. Marcell, M.M., Borella, D., Greene, M., Kerr, E., Rogers, S.: Confrontation naming of environmental sounds. *J. Clin. Exp. Neuropsychol.* **22**(6), 830–864 (2000)
30. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M.: The DET curve in assessment of detection task performance. In: *Proceedings of Eurospeech, Rhodes, vol. 4*, pp. 1895–1898 (1997)
31. Mesaros, A., Heittola, T., Palomäki, K.: Analysis of acoustic-semantic relationship for diversely annotated real-world audio data. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 813–817. IEEE Computer Society, Los Alamitos, CA (2013)
32. Mesaros, A., Heittola, T., Virtanen, T.: Metrics for polyphonic sound event detection. *Appl. Sci.* **6**(6), 162 (2016)
33. Mesaros, A., Heittola, T., Virtanen, T.: TUT database for acoustic scene classification and sound event detection. In: *24th European Signal Processing Conference 2016 (EUSIPCO 2016)* (2016)
34. Pallett, D.S.: A look at NIST’s benchmark ASR tests: past, present, and future. In: *2003 IEEE Workshop on Automatic Speech Recognition and Understanding, 2003. ASRU’03*, pp. 483–488. IEEE, New York (2003)
35. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6440–6444 (2016)
36. Paul, D.B., Baker, J.M.: The design for the wall street journal-based CSR corpus. In: *Proceedings of the Workshop on Speech and Natural Language, HLT ‘91*, pp. 357–362. Association for Computational Linguistics, Stroudsburg, PA (1992)
37. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: *IEEE International Workshop on Machine Learning for Signal Processing* (2015)
38. Piczak, K.J.: ESC: dataset for environmental sound classification. In: *Proceedings of the ACM International Conference on Multimedia (ACM)*, pp. 1015–1018 (2015)
39. Poliner, G.E., Ellis, D.P.: A discriminative model for polyphonic piano transcription. *EURASIP J. Adv. Signal Process.* **2007**(1), 154 (2007)
40. Rakotomamonjy, A., Gasso, G.: Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(1), 142–153 (2015)
41. Räsänen, O., Leppänen, J., Laine, U.K., Saarinen, J.P.: Comparison of classifiers in audio and acceleration based context classification in mobile phones. In: *19th European Signal Processing Conference (EUSIPCO)*, pp. 946–950. IEEE, New York (2011)

42. Rijsbergen, C.J.V.: *Information Retrieval*, 2nd edn. Butterworth-Heinemann, Newton, MA (1979)
43. Salamon, J., Bello, J.P.: Feature learning with deep scattering for urban sound analysis. In: *European Signal Processing Conference (EUSIPCO)*, Nice, pp. 729–733 (2015)
44. Salomon, J., Bello, J.P.: Unsupervised feature learning for urban sound classification. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, pp. 171–175 (2015)
45. Salamon, J., Bello, J.P.: Deep convolutional neural networks and data augmentation for environmental sound classifications. *IEEE Signal Process. Lett.* **24**(3), 279–283 (2017)
46. Salamon, J., Jacoby, C., Bello, J.P.: A dataset and taxonomy for urban sound research. In: *Proceedings of the ACM International Conference on Multimedia, MM '14*, pp. 1041–1044. ACM, New York, NY (2014)
47. Sammut, C., Webb, G.I.: *Encyclopedia of Machine Learning*, 1st edn. Springer, Berlin (2011)
48. Scott, W.A.: Reliability of content analysis: the case of nominal scale coding. *Public Opin. Q.* **19**(3), 321–325 (1955)
49. Sebastiani, F.: Machine learning in automated text categorization. *ACM Comput. Surv.* **34**(1), 1–47 (2002)
50. Sechidis, K., Tsoumakas, G., Vlahavas, I.: On the stratification of multi-label data. In: *Machine Learning and Knowledge Discovery in Databases. Lecture Notes in Computer Science*, vol. 6913, pp. 145–158. Springer, Berlin, Heidelberg (2011)
51. Shafiro, V., Gygi, B.: How to select stimuli for environmental sound research and where to find them. *Behav. Res. Methods Instrum. Comput.* **36**(4), 590–598 (2004)
52. Stowell, D., Plumbley, M.: An open dataset for research on audio field recording archives: freefield1010. In: *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio* (2014)
53. Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.: Detection and classification of acoustic scenes and events. *IEEE Trans. Multimedia* **17**(10), 1733–1746 (2015)
54. Takahashi, N., Gygli, M., Pfister, B., Van Gool, L.: Deep convolutional neural networks and data augmentation for acoustic event detection. In: *INTERSPEECH 2016* (2016)
55. Weiss, C., Arifi-Müller, V., Prätzlich, T., Kleinertz, R., Müller, M.: Analyzing measure annotations for western classical music recordings. In: *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)* (2016)
56. Yeh, A.: More accurate tests for the statistical significance of result differences. In: *Proceedings of the 18th Conference on Computational Linguistics*, vol. 2, pp. 947–953. Association for Computational Linguistics, Stroudsburg (2000)
57. Zieger, C., Omologo, M.: Acoustic event detection - ITC-irst AED database. Technical Report, ITC-irst (2005)
58. Zölzer, U. (ed.): *Digital Audio Signal Processing*, 2nd edn. Wiley, New York (2008)

Part III
Advanced Methods

Chapter 7

Everyday Sound Categorization

Catherine Guastavino

Abstract This chapter reviews theories and empirical research on the ways in which people spontaneously and effortlessly categorize sounds into meaningful categories to make sense of their environment. We begin with an overview of prominent theories of categorization in the psychological literature, followed by data collection and analysis methods used in empirical research on categorization with human participants. We then focus on auditory categorization, synthesizing the main findings of studies on isolated sound events as well as complex sound scenes. Finally, we review recently proposed taxonomies for everyday sounds and conclude by providing directions for integrating insights from cognitive psychology into the design and evaluation of computational systems.

Keywords Everyday sounds • Categorization • Cognitive psychology • Soundscape • Prototype theory • Linguistic labelling • Similarity • Taxonomies • Holistic perception • Top-down processes • Context

7.1 Introduction

The human ability to categorize is essential to make sense of the world by dividing it up into meaningful categories. Grouping together entities of the same kind serves to reduce the complexity of the environment. In our everyday life, we categorize things, objects, people, sounds, and situations continuously and effortlessly to infer further knowledge about them (e.g., what to do with them). Categorizing sounds in particular is of vital importance to handle the variety and complexity of complex environments and subsequently guide action (e.g., avoid an approaching car, attend to a crying baby, or answer a ringing phone). While speech sounds and music have been studied extensively, everyday sounds have long been under-investigated. Hearing research traditionally focused on artificial, synthetic sounds (e.g., pure tones or noise bursts) in controlled laboratory experiments. This was

C. Guastavino (✉)
School of Information Studies, McGill University, 3661 Peel street, Montreal, QC,
Canada H3A 1X1
e-mail: catherine.guastavino@mcgill.ca

partly due to the limitations of the available technology and instrumentation with regard to generating and analyzing complex dynamic sounds (see [44] for an historical perspective of technological developments in psychoacoustics research), and partly due to the dominant psychophysical approach to perception. This body of research has provided a better understanding of the functioning of the auditory system with a focus on lower level sensory processing. However, the sounds and listening conditions tested bear little resemblance to listening situations encountered in everyday life.

More recently, everyday sounds garnered increased research attention within the cognitive ecological approach to auditory perception as important components of everyday experiences. In contrast to artificial, synthetic sounds, everyday sounds are defined as sounds occurring in real-life environments [1]; they are also referred to as environmental sounds in the literature (e.g., [22, 29, 61]) or domestic sounds for everyday sounds typically heard inside the home [28]. From an evolutionary perspective, it can be argued that the ability to make sense of everyday sounds around us preceded speech and music abilities. Everyday sounds also provide a gateway into understanding intricate spectro-temporal structures (other than speech) and to study auditory perception in a broader context of cognition and action (e.g., how we give meanings to sounds and rely on sound to make sense of and interact with our environment). Research on everyday sounds has also been motivated by practical applications including:

- medicine and related areas (e.g., diagnostic tools for hearing impairments, hearing aids to restore environment awareness through sound);
- audio for computer-mediated environments (e.g., virtual reality, videoconferencing, video games, media arts installation);
- auditory comfort, product sound quality and soundscape design and assessment;
- and computational systems (e.g., acoustic monitoring, automated sound recognition and/or classification).

This chapter aims to review a body of research of particular significance to the current volume, that illuminates converging evidence for ways in which people spontaneously and effortlessly categorize sounds into meaningful categories. This area of inquiry is critical to inform the design and evaluation of sound recognition systems in terms of sound events that would be perceived as meaningful for listeners in particular situations. There is also insight and guidance to be had from the wider groupings and abstractions made by listeners: they can point to those features or correlates of attributes that are most important for categorization, and dictate the kinds of confusions and generalizations that will or will not be acceptable to users. We begin with an overview of prominent theories of categorization in the psychological literature in Sect. 7.2. Section 7.3 describes data collection and analysis methods used in empirical research with human participants to study categorization. Section 7.4 focuses on auditory categorization, synthesizing the main findings of studies on isolated sound events as well as complex sound scenes. Finally, in Sect. 7.5, we review some recently proposed taxonomies for everyday sounds and conclude by providing directions for making research-based connections between cognitive psychology and computational systems.

7.2 Theories of Categorization

7.2.1 *Classical Theory of Categorization*

While the classical theory of categorization goes back to the ancient Greeks, its influence has been pervasive and long lasting in psychology (and cognate fields such as philosophy, linguistics, information science) and prevailed throughout much of the twentieth century. In the classical approach of Aristotle, categorization relies on a conjunction of sufficient and necessary conditions [2]. The conditions are binary: an entity either possesses a feature or it does not. Category membership is also binary (an entity either belongs to a category or it does not, sometimes referred to as the all-or-none principle) and based on deduction: an entity is a member of the category if and only if it possesses all the defining features of the category. This theory establishes clearly delineated boundaries between categories. Categories are mutually exclusive and collectively exhaustive, and all members of a category have equal status. Although the classical theory of categorization was a philosophical position, arrived at through speculation rather than grounded in empirical evidence, it became an unquestionable assumption in many disciplines for centuries. In this view, categorization is conceived of as a deductive analytic process: one needs to identify features to determine category membership. While this analytic view is relevant for categories of biological kinds, abstract concepts, or artificial stimuli, it has been challenged by perceptual research on the categorization of concrete objects (e.g., furniture, vehicles) in the twentieth century. This line of research led to the idea that more holistic processes are at play in everyday categorization.

7.2.2 *Holistic Perception*

Holism is a central tenet of Gestalt psychology, which argues that we perceive perceptual objects as sensory wholes (Gestalts) rather than the sum of their parts. These wholes possess features that cannot be derived from their constituent parts and are structured using grouping principles based on similarity, proximity, common fate, and good continuation. While these grouping principles have thoroughly been investigated and modeled in the visual domain (e.g., face perception), they were originally motivated by auditory considerations. Wertheimer [62] discusses the emergence of a holistic perceptual object (a melody) in response to the sounding of disparate sound events (individual notes). The separation of melody and accompaniment in music listening motivated the formulation of figure-ground segregation. The need to explain the preservation of perceived melodic identity under key transposition inspired the grouping principles later named similarity and common fate, where similar sound events or sound events moving together (in pitch and time) are likely to be perceived as a perceptual object [16]. These principles inspired research on auditory scene analysis to determine how a sequence of sound

events are fused or segregated into different streams/units by generating boundaries between perceptual objects [5] (see Chap. 3 of this volume for a summary). In the field of philosophy, Wittgenstein introduced the notion of “family resemblance” to characterize the relationships of different instances of a category whose members have no one feature or set of features in common [64]. He exemplifies this notion with games, comparing different kinds of games (e.g., board games, card games, ball games) and concluding that they are related by a “complicated network of similarities overlapping and criss-crossing.” This view challenges the classical theory of categorization, specifically the all-or-none principle of category membership and the rigid boundaries between categories, and provides further support for holistic processes in categorization.

7.2.3 *Prototype Theory of Categorization*

The pioneering work of Eleanor Rosch constitutes a radical departure from the classical theory of categorization based on the study of natural categories (as opposed to artificial stimuli), bringing categorization to the central stage of cognitive science in the 1970s. Rosch’s landmark studies relies on categorization principles that go far beyond shared features [49, 50]. Within the Roschian model, categorization is not just deductive but also inductive, insofar as we can infer properties of an entity from observing characteristic features of its category members. Rosch proposes that a category members are related to one another through family resemblance and that certain category members can be more or less typical exemplars of that category (e.g., a robin is more typical of the category bird than a penguin). A family resemblance relationship exists when each category member has certain features¹ in common with one or more members, but no, or few, features in common with all members of the category. The most central member of the category is called a prototype; it is a member that best reflects the category, having more common features with other category members and least with members of other categories. Prototype theory posits that the internal structure of a category is based on similarity to a prototypical exemplar: category membership is defined by the extent to which a member of a category resembles the prototype. Rosch further formalized different levels of abstraction in categorization: superordinate (furniture, animal), basic (chair, bird), subordinate (office chair, blue jay) in a number of influential studies. These different levels are represented in Fig. 7.1. The basic level is at a middle level of specificity and contains relatively more information (lots of features) at a relatively low cost, making it the most natural, preferred level for identification and categorization. Given that something is a chair, you can predict

¹It should be noted that Rosch talks about attributes rather than features, but we use features as distinctive characteristics, properties, or quality for the sake of consistency in this chapter (although features itself has multiple meaning within this book, e.g., acoustic features in Chap. 4).

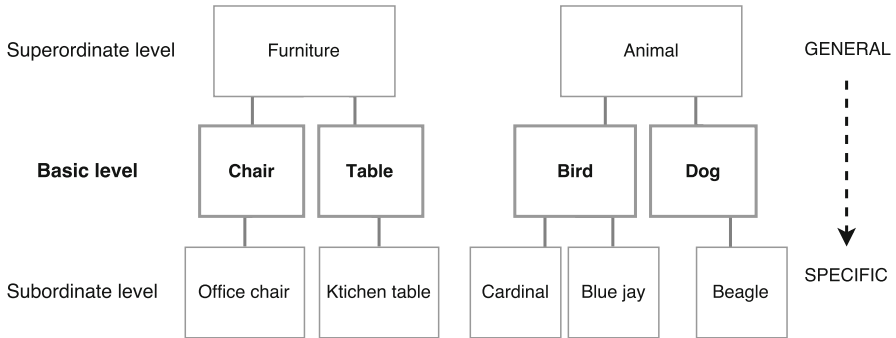


Fig. 7.1 Different levels of abstractions for natural categories (Rosch [49])

more about its appearance (it likely has a seat, a back and legs) and its function (used to sit on) than if you know only that it is a piece of furniture. Knowing that it is an office chair rather than a different type of chair would not change these prediction much, only further specify form or function. In Rosch's terms, the basis level maximizes cue validity, that is, the probability that a particular entity belongs to some category given that it possesses a particular feature, or cue. Basic level categories are both informative (knowing that it something is a bird, you can infer that it has feathers, a beak, etc.) and reasonably distinctive (different from other categories at the same level, e.g., cats). Subordinate categories are also informative (even more than basic categories) but less distinctive (many features in common, e.g., a blue jay is not very different from a cardinal). Superordinate categories, on the other hand, are very distinctive (e.g., furniture and tools do not have much in common) but not very informative (few features in common, e.g., different pieces of furniture can vary considerably in size, shape and functions). According to Lamberts and Shanks [35, p. 100], “the basic level can be seen as a compromise between the accuracy of classification at a maximally general level and the predictive power of a maximally specific level”. This basic level is associated with lexicalization. Basic category names are typically nouns and are used more frequently by adult speakers (e.g., in picture naming tasks). There is also developmental evidence that the basic level categories are the first acquired by children. The other levels (superordinate and subordinate) are harder to learn and to describe with names. In Sect. 7.3, we will discuss what these basic levels are in the context of everyday sound categorization.

7.2.4 Exemplar Theory of Categorization

Recent theories, including exemplar theory, posit that categorization relies on the comparison of a new entity with members (exemplars) of the category previously stored in memory (e.g., [47, 56]). Categorization is then determined by the degree of similarity between the new object and the stored exemplars, rather than a pro-

totypical exemplar (which can be seen as an abstracted category representation). In this view, an entity is categorized into the category with the most similar stored exemplars (rather than the category with the most similar prototype). This theory has met considerable success at predicting human categorization, but typically in the context of experiments using concrete objects (e.g., bird, chair) rather than more abstract concepts (e.g., beauty, the number five) which require abstracting essential information from particular instances. In addition, there is evidence that the two theories are not mutually exclusive and various studies have investigated conditions under which categorization will be guided by prototype or exemplar theories. Although there are differences between prototype and exemplar theories, they both consider that categorization is based on the similarity between the entity to-be-categorized and the category representation. These theories account for the flexibility and plasticity of the cognitive process of categorization and provide grounds for probabilistic models of category membership with graded structure and category boundaries that are not sharply delimited.

7.2.5 Bottom-Up and Top-Down Processes

Under most recent theories, categorization is achieved using a mixture of bottom-up processes triggered by features of the signal and top-down processes based on expectations and previous knowledge. Bottom-up processing goes from the low level of sensory registration to higher cognitive levels. These processes (a.k.a. data-driven or stimuli-driven), directly influenced by features of the stimuli, serve as grounds for the information-processing approach to human perception. Conversely, top-down processes (a.k.a. concept or theory or hypothesis-driven) shape sensory registration based on an individual's expectations, prior knowledge, and contextual factors. In this view, perception is driven by high-level cognitive processes as opposed to strictly processing incoming signals. Many models of identification and categorization initially relied on a serial account of processing where bottom-up processing needs to be completed at a first stage before it is passed on to higher levels of processing. However, there is converging evidence from behavioral and neurophysiological studies on object identification [31] as well as speech perception providing support for an integrated model of bottom-up and top-down processing² rather than a serial model. This is also consistent with current computational approaches, in which the interaction of low-level features with high-level constraints is blurred. Speech recognizers, for example, neither attempt to fully transcribe into phonemes before matching to words, nor to exhaustively enumerate all possible word sequences before rating the presence of phonemes.

²Based among other things on the evidence for top-down activation in early stages of visual processing (e.g., for figure-ground segmentation).

Bottom-up processes rely on the similarity between items while top-down processes rely on the theories³ people hold about the world. Goldstone reviewed evidence that categorization cannot be accounted for completely by similarity as it involves properties that are not obtainable from only individual item similarity [20]. Even the notion of similarity is context-dependent, as many researchers observed, when manipulating the context of a similarity comparison. In particular, similarity is dependent on the context defined by the stimulus set (e.g., Tversky, 1977), on cultural context (e.g., [10, 63]), on the level of expertise and goals of the subjects (e.g., [60]) and on the presentation context of the item. For example, Medin and Shoben found that similarity between adjectives is influenced by the accompanying noun [41]. In their study, white was selected more often than black as being similar to grey when accompanying the noun hair, and the opposite trend was observed when the adjectives were accompanying the noun clouds.

In the context of everyday perception, person-related factors (e.g., prior knowledge), as well as situational factors, play an important role in categorization. In the absence of established shared knowledge of the categories, categorization principles rely mostly on experiential knowledge. Barsalou investigated the construction and use of goal-derived ad hoc categories (e.g., “places to go on vacation”) [3]. He found that entities categorized using basic category names (e.g., “chair” as part of “furniture”) can be cross-classified in other situations that serve particular goals (e.g., as part of “things that can be stood on”). Dubois argues that the influence of prior knowledge on categorization refers not only to individual knowledge but rather knowledge grounded in shared socialized activities [14]. As an example, the category “things to bring to a birthday party” will be influenced by a number of social factors determining what would be appropriate in a given context based on envisaged activities, age group, relationship to the guests, etc. Similarly in the auditory domain, honking car horns could be associated with a celebratory gesture for a wedding party in certain social contexts, or to traffic jams or road crossing in other contexts. To mediate between individual experience and shared knowledge, the analysis of free-format verbal descriptions provides relevant insights since language is by essence both shared and individual.

Also, research on food categories revealed organization principles based on situational factors. Ross and Murphy observed the salience of script categories, i.e., categories referring to the situation or time in which the food is eaten (e.g., foods to eat at breakfast time) [51]. It was further shown that food could be cross-classified either into taxonomic categories on the basis of similarity (foods of the same constitutive kinds, e.g., “vegetables”) or into script categories on the basis of human interactions (e.g., “breakfast foods”). Script categories can be of importance to generate plans in larger goal-oriented tasks (e.g., deciding what to eat). Most importantly, cross-classification highlights the co-existence of different categorization principles for everyday objects.

³Here, theory is understood to mean any organized system of knowledge, “folk” as well as scientific theories.

This cross-classification into taxonomic and script categories has been observed in adults but also in children as young as 3 years old [45], indicating that even at a young age, children do not rely on a single form of categorization but are flexible in the types of categories they form and use in everyday life.⁴ This suggests that cross-classification is an essential ability to get a full understanding of the world around us (e.g., being able to think of someone as a man, a father, a friend, a music lover or a Frenchman can be useful to understand the complex behavior of a person). The extent to which these different aspects contribute to everyday sound categorization will be discussed in Section 7.4. In Section 7.3 we first describe the data collection and data analysis methods most commonly used in listening tests investigating sound categorization.

7.3 Research Methods for Sound Categorization

This section provides an overview of data collection methods and data analysis techniques used in categorization studies of sound stimuli with human participants. As discussed above, similarity is a central construct to study categorization, even if it does not tell the whole story (discussed below). Empirical studies on categorization often aim to model similarity as a function of the features of the presented stimuli. We first present methods most commonly used to gather similarity judgments.

7.3.1 Data Collection

7.3.1.1 Dissimilarity Estimation

Among the various methods for auditory research, dissimilarity estimation is perhaps the most widely used in the context of psychoacoustics, a field that has mainly developed along the psychophysical tradition of comparing “subjective” perceptual judgments with “objective” description of stimuli in terms of their physical properties. Dissimilarity ratings are collected for each of the $N(N - 1)/2$ pairwise combinations of N stimuli. On each trial, participants rate how similar or different two stimuli presented in paired comparison are, along on a scale. The scale can be discrete or continuous; the end points are sometimes labeled “very different” and “very similar.” Each pair is presented twice in counterbalanced order and the order of presentation across trials is randomized to nullify order effects. This method is appropriate for homogenous data sets, or to determine how sensitive listeners are

⁴Children also use thematic categories of entities formed on an associative basis (e.g., dog and leash). Members of thematic categories are not similar and do not share many features but they are often spatially and temporally contiguous and play complementary roles (contrary to members of script categories which play the same role).

to a particular dimension of the acoustic signal. But it orients the listeners' strategies toward adopting an analytical approach with ratings on a single dimension, in response to the manipulation or selection of stimuli introduced by the researcher. This method also requires a large number of trials (increasing quadratically with the number of sounds tested), which makes it prone to fatigue or boredom effect with large data sets.

7.3.1.2 Sorting Tasks

As an alternative to dissimilarity estimation, sorting methods rely on participants creating groups of similar sounds. Different variants of the sorting task exist. The number of groups could be pre-defined by the experimenter (fixed sorting) or left up to the participants (free sorting). The categorization principle can be pre-defined by the experimenter (closed sorting, e.g., "group sounds produced by the same object") or decided on by participants (open sorting). In an open, free sorting task (a.k.a. free categorization task), participants are presented with N sounds and asked to create groups of sounds that belong together. Participants are free to decide how many groups they want to create and on which criteria they group sounds together. While open free sorting tasks rely on similarity judgment, they also involve more holistic decisions and do not restrict the strategies used by participants. This method is appropriate for exploratory studies with heterogeneous data sets to identify relevant features or correlates of attributes along which participants spontaneously organize stimuli in the absence of a priori assumptions regarding the number of categories. It is also relevant to identify different categorization principles across participants as well as for a given participant across different subsets of sounds. Furthermore, it allows researchers to test a fairly large number of sounds in a relatively short amount of time. Other methods for collecting similarity ratings include hierarchical sorting tasks in which participants start with sounds in different groups and merge the two most similar groups or sounds in a recursive manner until all sounds are grouped together. A detailed comparison of the methods of dissimilarity ratings, free sorting, and hierarchical sorting in terms of efficiency, reliability, and accuracy can be found in [19].

7.3.2 Data Analysis

Similarity ratings can be summarized in the form of a dissimilarity matrix Δ . An individual matrix is then generated for each participant. The value in the i th row and the j th column of the dissimilarity matrix Δ , denoted ∂_{ij} , is defined as follows:

- $\partial_{ij} = 0$ if i and j are in the same category,
- $\partial_{ij} = 1$ if i and j are not in the same category.

A global dissimilarity matrix can be obtained by summing the individual dissimilarity matrices.

7.3.2.1 Multidimensional Scaling

Multidimensional scaling (MDS) is a set of mathematical techniques to represent the similarity between N items (here the global dissimilarity matrix) in terms of fewer dimensions in a way that best approximates the observed similarity. Each item is represented as a point in that space. Similar items are represented by points that are close in space while dissimilar items are represented by points that are far apart. The space is usually a 2-D or 3-D Euclidian space, but different distances (metric, non-metric) and more dimensions can be used. The goodness of fit of the representation can be estimated using metrical criteria (scree and percentage of explained variance). The goodness of fit increases with the number of dimensions. The scree plot (representing stress as a function of dimensions) can be used to determine the optimal number of dimensions. The analysis aims to identify underlying attributes explaining similarity judgments. The interpretation of the underlying attributes relies on the visual inspection of spread and clusters in the scatterplots and on the use of multiple regression techniques [34] to interpret these dimensions of the stimulus space in terms of similarities between items. MDS also provides models to represent individual differences as weights on each underlying dimensions [a.k.a. as weighted model or Individual Difference scaling (INDSCAL)]. An example of a 2-D MDS representation is shown in Fig. 7.2.

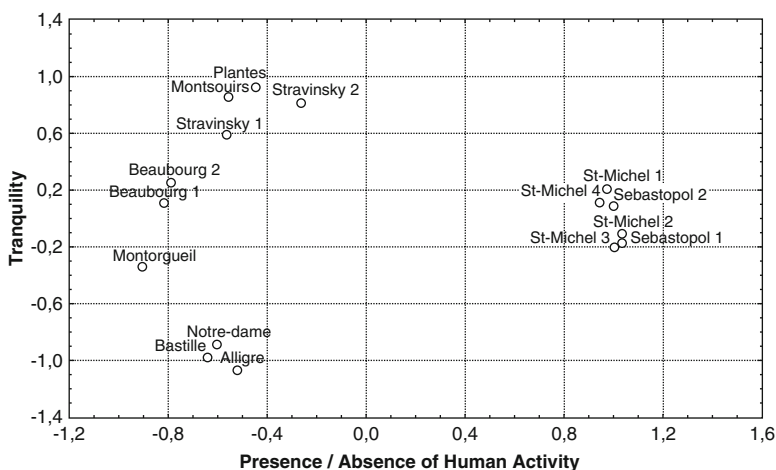


Fig. 7.2 The two-dimensional representation of the data derived from MDS analysis of a dissimilarity matrix (adapted from [22]). The dissimilarity between two objects is represented by the Euclidean distance between the two corresponding points in space

7.3.2.2 Additive-Tree Representations

Additive-tree representations (a.k.a. phylogenetic trees) are used in a variety of disciplinary fields, ranging from computer science to biology, as a graphical representation of dissimilarity data. In cognitive psychology, categorization research has been a fruitful source of inspiration for the additive tree theory. Rosch’s prototype theory initiated the development of a new theory of psychological similarity, which can be represented by additive trees [53]. Additive trees were designed to account for several empirical observations, including the fact that some members are more typical of a category than others. Indeed, the traditional taxonomic tree representation, in which all items are at the same distance from the root, forces all members of a category to be equivalent. The additive tree representation, with edges of varying lengths, seems more appropriate to represent a gradient of typicality. Formally, an additive tree is a connected, non-directed, and acyclic graph, together with an additive distance. The items are represented by the “leaves” (or terminal nodes) of the tree, and the observed similarity between items is represented by the distance between leaves along the edges. The goodness of fit is expressed in terms of both metrical criteria based on edge lengths (stress, percentage of variance explained), and topological criteria based on the tree topology (arboricity, percentage of well represented quadruplets) [27]. An example is shown in Fig. 7.3.

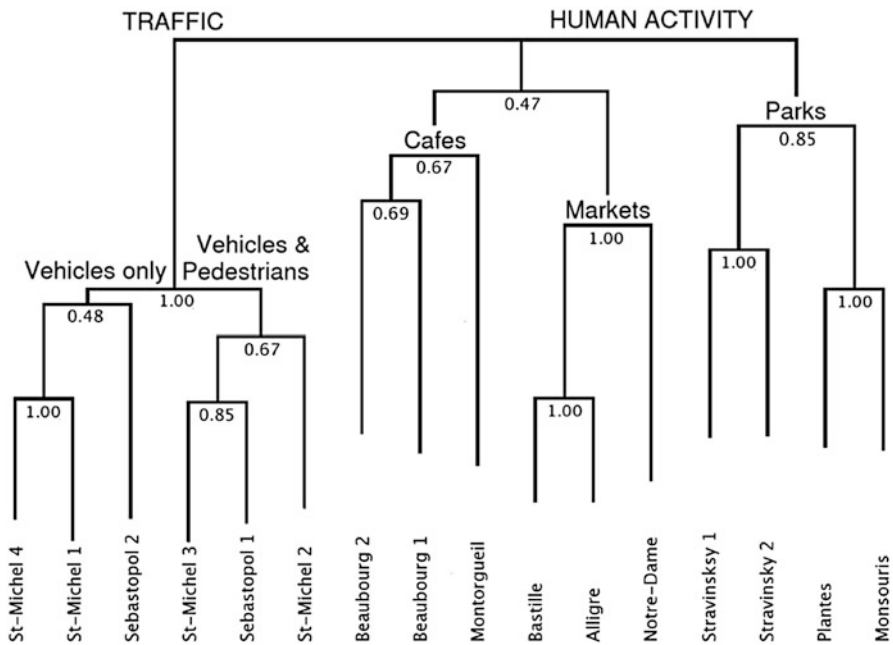


Fig. 7.3 Additive tree representation of the same dissimilarity matrix with verbal descriptors of the main categories (adapted from [22]). Here, the dissimilarity between two objects is proportional to the length of the edge path connecting them in the tree. The number between 0 and 1 shown at each node is a topological indicator of goodness of fit of a given edge. The greater this number is, the more reliable the grouping between the corresponding nodes is

7.3.2.3 Mantel Test

In order to compare dissimilarity matrices directly (e.g., between two groups of participants or two experimental conditions), the Mantel test can be used for significance testing [39]. A straightforward correlation analysis of two distance matrices cannot be carried out because the distances in the matrices are not independent. The Mantel test provides a way to overcome this difficulty. The rationale behind the Mantel test is that, if there is no correlation between the matrices, then a random permutation of their rows and columns will produce equally likely low correlation coefficients. Thus, the test performs random permutations of the rows and columns of the matrices and computes the normal correlation coefficient; after that, it counts the proportion of those permutations that led to high correlation coefficients. A hypothesis testing is then computed to determine the final correlation of the distance matrices (see [26] for an application of the Mantel to test for differences between groups of participants evaluating rhythmic similarity).

7.4 How Do People Categorize Sounds in Everyday Life?

We present in this section the main findings of selected behavioral studies on everyday sound categorization. We first examine studies using isolated sound events and then studies investigating complex auditory scenes.

7.4.1 *Isolated Environmental Sounds*

In a seminal study, Vanderveer conducted a free sorting task and a free identification task of recordings of isolated everyday sound events, asking participants to group and describe each sound in their own words [61]. Participants described and organized the sounds in terms of sound source, i.e., the object producing sound or the action generating the sound (e.g., “tearing paper”). They referred to descriptions of the acoustic signal only when they could not identify the sound event. These findings provide support for Schubert’s view that “identification of sound sources and the behavior of those sources is the primary task of the auditory system” [55]. Gaver argued for an ecological approach to auditory perception and introduced the distinction between musical listening and everyday listening [17]. Musical listening focuses on perceptual attributes of the sound itself (e.g., pitch, loudness), whereas everyday listening focuses on events to gather relevant information about our environment (e.g., car approaching), that is, not about the sound itself but rather about the source and actions producing the sounds, and what they might mean. Gaver further explains that a given acoustic phenomenon can give rise to both modes of listening depending on whether the listener focuses on the properties of the sound itself or rather on the event that caused the sound: “The distinction between

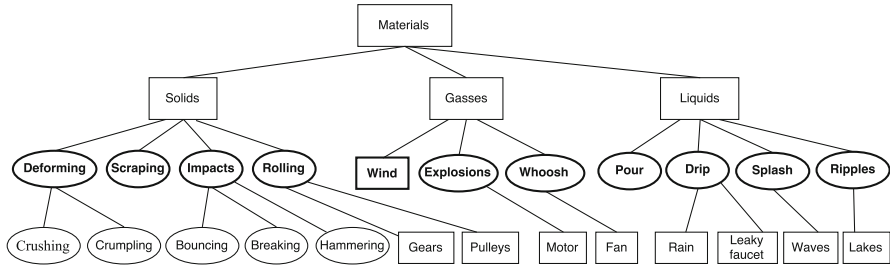


Fig. 7.4 Gaver’s taxonomy of sound everyday sounds (adapted from [17]’s Figs. 6 and 7). Basic level shown using a *thicker outline* and *square boxes* for represent sound sources, *circles* represent actions

everyday and musical listening is between experiences, not sounds” [17, p. 1]. Gaver also proposed a taxonomy of sound events based on the physical description of sound production. Sound events are organized, first, in terms of interacting materials at a higher level, then sound producing events (three categories: solids, gasses and liquids), and finally interactions that cause sound (e.g., impact, rolling). This taxonomy is represented in Fig. 7.4 (adapted from [17]). Gaver also mentions hybrid events involving more than one sort of material (e.g., rain on a surface). It should be noted that his taxonomy includes sound sources and actions. For the sake of clarity, in all the figures in this chapter, we represent sources with square boxes and actions with circles. When a basic level of categorization is explicitly mentioned by the authors, we use a thicker outline for the boxes representing the basic level.

Guyot et al. conducted a free sorting task of recordings of domestic sounds and asked participants to describe each category and the relationships between categories [28]. Similar to Vanderveer [61], they observed two categorization principles, one based on sound source similarity and the other based on the similarity between event, or action, causing the sound. It should be noted that the same acoustic phenomenon could be categorized either based on action generating noise (e.g., “squeaking”) or based on the sound source (e.g., “door sounds”). Relying on psycholinguistic analyses within the Roschian categorization framework, the authors proposed a hierarchical organization of everyday sounds. For sounds resulting from a mechanical excitation, the basic level (shown in thick outlines in Fig. 7.5) is represented by actions generating sounds (e.g., “scratching”, “rubbing”, shown in round boxes), while the subordinate level represents sound sources (e.g., “dishes”, “Velcro”, shown in square boxes) or correlate of sources and actions (e.g., “pen sharpening”, rounded square boxes). The superordinate level represents a higher level of abstraction in terms of sound production as mechanical vs. electrical.

Marcell et al. [40] and Gygi et al. [29] extended this investigation to large collections of sound events. Their findings confirmed evidence for categorization principles based on source identification and event producing sound but also highlighted principles based on situational factors such as the location (kitchen, office) or context (sports) in which the sounds would be heard, as well as

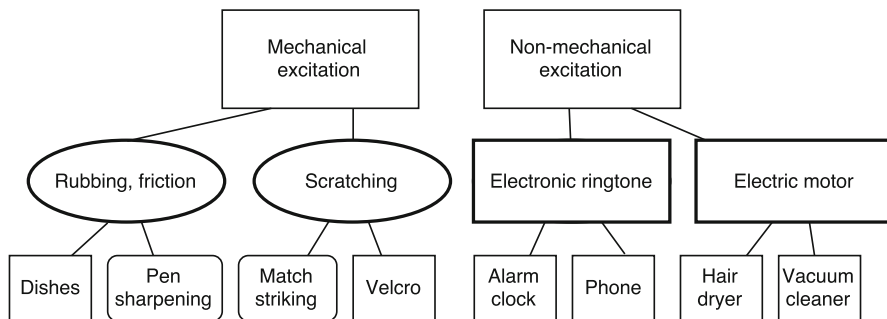


Fig. 7.5 Categories of domestic sounds adapted from [28]. Basic level shown in *thick outlines*, *square boxes* represent sound sources, *circles* represent actions, *rounded squares* for combinations of source and action

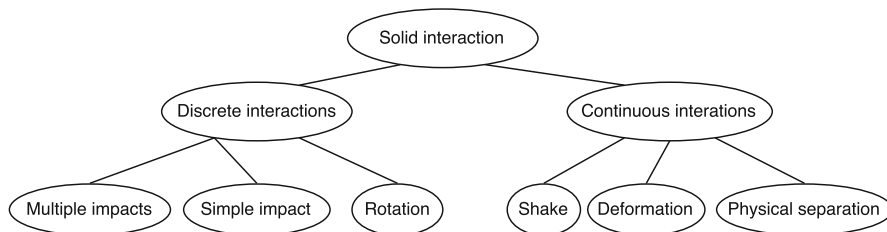


Fig. 7.6 Taxonomy of sound-producing actions adapted from [30]

hedonic judgments associated with emotional responses (e.g., annoying). In [30], Houix et al. validated Gaver's taxonomy through free sorting tasks and lexical analysis of verbal descriptions. In a first free sorting task using a heterogeneous set of (kitchen) sounds, participants based their categorization on the different types of sound sources, namely solids, liquids, gasses, and machines. In a second experiment using only sounds produced by solids (in an indoor environment), participants were asked to organize sounds according to the action generating sounds. The authors observed a distinction between discrete interactions (e.g., impact) and continuous interactions (e.g., deformation) as shown in Fig. 7.6. But they observed fewer categories of actions than proposed by Gaver [17]. These results suggest that the set of actions is constrained by the type of objects, suggesting a close interaction between action and sound source. This can be contrasted with Gaver's view that action and sound source are independent categorization principles.

Furthermore, these different studies indicate that sounds can be cross-classified according to different categorization principles depending on the context of presentation and the participants' goals and theories. The extent to which participants rely on acoustic properties of the sound vs. semantic properties of the sound source also varies for different types of sounds. For example, Giordano, McDonnell and McAdams found that participants rely more on semantic properties of the source for sounds produced by living agents (referred to as animate sounds) and more on

acoustical properties of the sound itself for sounds produced by non-living agents (referred to as inanimate sounds) [18]. This is also reflected in linguistic resources: while there are lexical forms for sound produced by humans (e.g., voice, laughter, footstep, burp), there are few single words on which people agree as spontaneous descriptions of sounds [11, 14, 15].

7.4.2 Linguistic Labelling of Auditory Categories

Indeed, most natural languages reflect conceptualizations of everyday sounds as individual experiences rather than shared knowledge, even in specialized industries focused on sound. The absence of collective norms and negotiated meaning of lexical forms for everyday sounds can be contrasted with firmly shared meanings for visual objects [9]. In the visual domain, categories of colors and shapes are elaborated as abstract categories autonomously from the colored object. As an example, one can easily think of the color red without having to think of a red object, that is, the color property (red) can be abstracted from individual instances of colored entities (a red apple, a red car or a red shirt). This is reflected in discourse with lexical forms for visual properties such as colors (e.g., red, green, blue) and shapes (e.g., circle, square, triangle). However in the auditory domain, categories are primarily structured based on the sources (object or agents) producing sound (e.g., car sounds, baby crying). In that sense, they are less abstracted from the source (e.g., car, baby) and are conceptualized as indicating the presence of an object or an agent and the effect it has on the listener. Discourse analyses conducted on free descriptions of everyday sounds reveal a large variety of linguistic devices. From *what* is being said and *how* it is being said, psycholinguistic analysis can be used to derive inferences about how people process and conceptualize sensory experiences [14]. This analysis mediates between individual sensory experiences and collective representations shared in language and elaborated as knowledge. The most frequent phrasings (see [14] for a review) spontaneously used to describe everyday sounds are:

- denominations constructed with a generic term (*noise, sound*) and a noun referring to the source (*door sounds, sound of a car*),
- suffixed nouns derived from verbs referring to the action generation the noise (*braking of a car*)
- adjectives derived from verbs referring mostly to hedonic or emotional factors (*pleasant, unbearable*) suggesting that sounds are conceptualized as effects of a world event on the listener (see [12] for an interpretation of the different suffixations in French).

The lack of basic lexicalized terms or a priori established categories questions the relationship between words and knowledge representations and suggests broadening the analysis beyond single words to complete statements provided by language in discourse.

7.4.3 Factors Influencing Everyday Sound Categorization

Over the past decade, there has been a growing body of literature of the categorization of everyday sounds. This section aims to synthesize previous research in terms of factors that have been found to influence categorization.

Similarity plays a critical role in categorization, but different types of similarities have been proposed in [36] based on a review of the literature.

- *Acoustical similarity* of the acoustic phenomena (see Chap. 3 for a review of the psychoacoustics of sounds scenes and events)
- *Causal similarity* defined as “the similarity of the identified physical event causing the sound” [36, p. 18], often reflected in the temporal structure of the sound (e.g., [30]).
- *Semantic similarity*, which refers to the meaning attributed to the sounds. This can be based on situational factors (locations and context in which sounds occur, referred to as *context-based* similarity, e.g., kitchen sounds) or based on the semantic properties of the sound source (referred to as *source-based* or *object-based* similarity) or the agent producing sound.

These different principles of categorization can operate together. Indeed, Morel et al. investigated road traffic noise and found that participants elaborated categories of sounds that combined the type of vehicle (e.g., car, truck, motorcycles) as well as the driving conditions (e.g., acceleration, deceleration) causing sound [42]. Furthermore, in a study using recordings of cylinders of different sizes and material (wood, plastic, glass, metal) undergoing different actions (scraping, rolling, hitting, bouncing), material perception was found to be fragile across different actions [36]. Participants were able to identify both the object (in term of size and material) and action producing sounds, but that they were always more accurate and faster at identifying actions.

Furthermore the influence of *person-related factors* on everyday sound categorization has been demonstrated. These include:

- *Expertise of the listeners*: Guyot et al. found that acousticians freely categorized sounds according to the properties of the acoustic signal (e.g., pitch, temporal evolution), while non-acousticians based their categorization on the sound sources or the actions generating sound [28]. The linguistic analysis of free-format verbal descriptions indicates that acousticians conceptualize sounds as abstract acoustic phenomena (i.e., as a perceptual object in itself), whereas non-acousticians conceptualize sounds as indicating the presence of an object that is not abstracted from the sound source. Lemaitre et al. further investigated the effect of prior knowledge by contrasting “expert” listeners (defined as musicians, sound artists, acousticians, or sound engineers) and untrained listeners and found similar results [36]. Similarly, expert listeners tended to categorize sounds on the basis of acoustical similarity,
- *Age*: Berland et al. conducted a developmental study of everyday sound categorization based on free-sorting tasks with young children, teenagers, and

adults [4]. They found that all three groups tended to rely on semantic similarity suggesting that the bases for this form of sound categorization are present in childhood. However the authors observed that several strategies could co-exist within a given development stage and that younger children created more script categories than adults.

- *Preferences*: Hedonic judgments have been found to be relevant and even discriminant to sort out unpleasant sounds from neutral and pleasant ones. In this case, categorization operates in relation to the perceived *identity* of the object/agent producing sound based on memory representations.

Situational factors such as mood, attention paid to the sound, and the activity carried out by a person while hearing the sound have had demonstrated modulating effects on auditory judgments. These findings have been reported in the context of sound quality evaluation of product sound and complex sonic environments (see [59]). While their effect on categorization has not been formally investigated (e.g., using free-sorting tasks or dissimilarity ratings), we can speculate that they would influence categorization since categorization relies on hedonic judgments.

7.4.4 *Complex Auditory Scenes*

A growing body of literature in the field of soundscape research has shed light on the cognitive and perceptual mechanisms people use to sort out mixtures of sounds into discrete categories in their everyday lives. The notion of soundscape was introduced by Schafer in the context of acoustic ecology in the 1970s [54] and has grown to connect with other related fields like community noise, acoustics, and psychoacoustics. Soundscape research emerged as a field in the late 1990s and the community of researchers working on urban soundscapes is particularly active in Europe and Asia. While different definitions had been proposed earlier, a recently formed ISO working group defines soundscape as “the acoustic environment as perceived or experienced and/or understood by a person or people, in context” [33]. This view emphasizes the importance of person-related and situational factors in everyday listening.

7.4.4.1 *Categories of Soundscapes as “Acts of Meaning”*

In her seminal paper [14], Dubois discusses auditory categories as “acts of meaning” extending Bruner’s conviction “that the central concept of a human psychology is meaning and the processes and transactions involved in the construction of meaning” [7, p. 33] to the auditory domain. This framework combined with Rosch’s prototype theory of categorization laid the theoretical and methodological groundwork for a cognitive approach to everyday sounds as meaningful events [15] in close relationship to linguistic labelling. What are these categories of soundscapes and how are they conveyed in language?

Maffiolo et al. investigated memory representations of familiar urban soundscapes with open questionnaires and mental maps [38]. The analysis of verbal and graphical descriptions made by city users suggests that soundscapes are structured into complex script categories integrating notions of time, location, and activities. These notions are reflected in discourse by complex prepositional phrases with multiple complements such as “riding motorcycles at Bastille on Saturday night”. In a similar vein, Guastavino analyzed free-format descriptions of familiar and ideal urban soundscapes [21]. The main categories of sounds identified were human sounds, traffic noise, natural sounds, and music. They were described in relation to hedonic judgments spontaneously evoked by respondents. Human and natural sounds gave rise to positive judgments (except when reflecting anger), whereas mechanical sounds gave rise to negative judgments. This distinction was even observed within certain categories such as music, which gives rise to two opposite qualitative evaluations depending on whether it reflected human activity directly (“musician”) or indirectly (“loudspeakers,” “car radio”). In the first case, it is perceived as lively and pleasant; in the latter, it is perceived as intrusive and therefore annoying. As regards mechanical sources, only electric cars and public transportation noise gave rise to positive judgments, in relation to environmental concerns. The evaluation of acoustic phenomena is therefore closely linked to the appraisal of the sound source and the meaning attributed to it, highlighting the importance of semantic features in categorization.

At a higher level of abstraction, results from free sorting tasks [37] and psycholinguistic analyses of verbal descriptions [23, 24] highlight a first distinction between *sound events*, attributed to clearly identified sources, and *ambient noise*, in which sounds blur together into collective background noise. *Sound events* are spontaneously described with reference to specific sources, by nouns referring to the object (*truck, bus*) or part of the object (*engine, muffler*) generating the noise. These metonymies—substituting the name of the source producing sound for the name of the sound itself—indicate confusions between sounds and sources producing the sound, and further suggest that the acoustic phenomenon is not abstracted from the object generating the sound. On the contrary, in the descriptions of ambient noise, there are few references to the object source and a majority of simple adjectives referring to the physical features of the acoustic signal (namely temporal structure and spectrum), suggesting a more abstracted conceptualization of the sound itself (that is, as a perceptual object in itself rather than as indicating the presence of an object in the world).

Finally, the comparison of verbal free-format description collected in actual environments and in laboratory experiments indicates that the sense of spatial immersion contributes to the cognitive representation of urban *ambient noise* [23, 24]. Guastavino et al. further showed that a multichannel surround sound reproduction, providing a strong feeling of immersion compared to low-channel setups, was necessary to ensure that urban noise reproduced in a laboratory setting is processed and subsequently evaluated in a similar manner to everyday life situations [25]. At a more generic level, Guastavino identified two main categories of urban soundscapes in a free sorting task, based on the perceived absence or

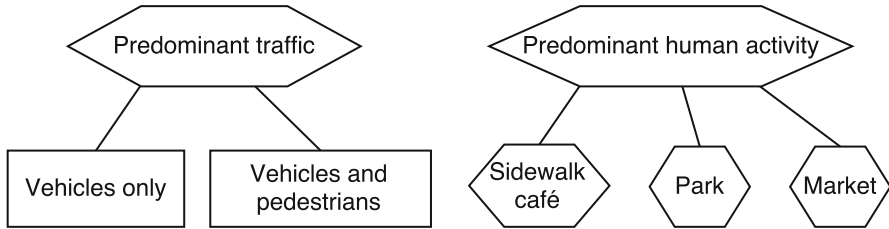


Fig. 7.7 Categories of soundscapes adapted from [22] corresponding to the representation shown in Figs. 7.2 and 7.3. *Hexagons* for activities/contexts, *square boxes* for sound sources, *circles* for actions

presence of human activity in the sound recording in relation to judgments of pleasantness [22]. Soundscapes in which traffic sounds dominated were tightly grouped together. They were subcategorized at a subordinate level according to sounds sources (the type of vehicles producing sound). Soundscapes in which human sounds dominated subdivided into subcategories related to the different types of activities ranging from busy markets to quiet parks (see Fig. 7.7). These findings provide support for another form of interaction between categorization principles based on sound sources, activities, and hedonic judgments. Furthermore, these findings are in line with the distinction between animate and inanimate agents observed with isolated everyday sounds [18].

A recent line of research further investigates the relationship between soundscape evaluation and activity. These evaluations rely on cultural values attributed to the different types of activities rather than on inherent properties of the sounds produced. Nielbo et al. asked participants to rate how appropriate different soundscapes were for different envisaged activities (e.g., studying, meeting with a friend) [46]. Findings showed that some of the tested soundscapes were rated as appropriate for all of the activities, some appropriate for no activities and a few of the soundscapes were appropriate for some of the activities but not others. Conversely, Steffens et al. collected soundscape evaluations “in situ” using the experience sampling method [58, 59]. Participants were prompted 10 times per day by a smart phone application to evaluate their soundscape and report on situational factors. The authors found the activity had a significant effect on soundscape evaluation: soundscapes were rated as more pleasant during recreational activities than during commuting or shopping. The effect could not simply be explained by different locations. A separate analysis of ratings collected in home environments also shows a significant effect of activity on soundscape evaluation, with recreational or entertainment activities associated with higher soundscape pleasantness than work, study, or other personal activities.

Other findings indicate that sounds contribute to the sense of place and encourage activities appropriate for the environment (e.g., marketplace sounds encourage conversation and purchasing). In addition, the context in which sounds are experienced plays a critical role in their evaluation. As an example, a study in a large pleasant park found that participants reported airplane noise as neutral rather than the

negative rating is usually attracts in studies conducted within the home [57]. Similar evidence that soundscapes provide relevant information about possible interactions with the environment comes from studies on auditory comfort evaluation inside passenger trains [43]. Passengers' auditory judgments were collected during a train ride using open-ended questions. Respondents evaluated sounds in relation to specific activities they were involved in. The evaluation of physical properties of sounds (e.g., sound level) was modulated by the activities of respondents. For instance, a given sound could be judged as “quiet enough to sleep” but “too loud to have a discussion.”

Together these findings suggest that the activity of the participants (task at hand or intended interaction with the environment) is a determining factor for everyday sound categorization and potentially even the factor that should be considered first.

7.5 Organizing Everyday Sounds

We now review some systematic classification schemes that have been proposed to account for the categorization of everyday sounds, and the extent to which they are informed by behavioral studies. Classification schemes can include either taxonomies and ontologies. Taxonomies are hierarchical structures of entities. The only relationship between entities in a taxonomy is class/subclass (a.k.a. as broad/narrow or parent/child relationship) based on class inclusion (e.g., animal kingdom taxonomy). Ontologies on the other hand model a broader range of relationships between entities.

7.5.1 Taxonomies of Sound Events

Murray Schafer proposed three different classification schemes for sound events, according to physical characteristics (duration, frequency, fluctuations, dynamics), according to aesthetic qualities and according to referential aspects [54]. These referential aspects refer to categories of sound sources and functions, described as

- *natural sounds* (e.g., sounds produced by water, animals, fire)
- *human sounds* (directly produced by humans, e.g., voice, footsteps)
- *sounds and society* which refers to the sounds of human activities (e.g., ceremonies) or different types of environments (city, domestic)
- *mechanical sounds* (e.g., machines, transportation)
- *quiet and silence*
- *sounds as indicators* (sounds that serve a particular informative function, e.g., warning signals, bells)

In [13], Delage proposed a classification of urban sounds according to the degree of human activity, with three classes of sounds:

- *sounds not produced by humans* (e.g., sounds of nature)
- *reflecting human activity indirectly* (e.g., traffic or construction noise)
- *reflecting human activity directly* (e.g., voices, footsteps)

In the context of soundscape ecology in natural ecosystems such as large parks and reserves, Pijanowski et al. proposed to categorize sounds into *geophony* (sounds from the geophysical environments such as rain and wind), *biophony* (sounds produced by biological organisms), and *anthrophony* (sound produced directly or indirectly by human activity) [48].

So far, the classification schemes reviewed are categories of sound sources. While these categorizations were not informed by empirical studies with listeners, similar principles have been observed in the results of empirical studies with isolated sound events as well as auditory scenes. However these proposed classification schemes do not account for the structure of sound events at different levels of abstraction (general to specific).

Furthermore as discussed in Sect. 7.4, different categorization principles co-exist, particularly in terms of sound sources and actions producing sounds. Salamon, Jacoby and Bello proposed a taxonomy of urban sounds that incorporates to some extent the action causing the sound [52]. At a superordinate level, the four categories are human, nature, mechanical, and music (a combination of categories of sources and sound production mechanisms). At lower levels, the leaves of the taxonomies represent categories of sound sources derived from the content analysis of noise complaints (filed through New York City’s 311⁵ service between 2010 and 2014). Some of these sources are associated with different actions (e.g., under Engine: idling, decelerating, accelerating). This is in agreement with the typology of vehicle sounds elaborated in [42] on the basis of free sorting tasks revealing two criteria, namely vehicle type and vehicle driving condition. In Fig. 7.8, we redraw a subset of their taxonomy adapted to clearly indicate sound sources (using square boxes), actions (using circles) or combinations of the two (rounded square boxes).

7.5.2 Taxonomies of Complex Auditory Scenes

Brown et al. proposed a categorization of soundscapes, with different categorization principles at different levels of specificity [6]. The superordinate level operates in terms of the type of environment (first distinction between indoor vs. outdoor, then between different types of environments within each), followed by a distinction in terms of presence or absence of human activity at the basic level, and then a classification in terms of sound sources at the subordinate level, as shown in Fig. 7.9.

The proposed distinction between different types of environment corresponds to different areas of expertise (soundscape researchers tend to specialize in a specific

⁵This special telephone number provides access to non-emergency municipal services (comments, complaints, questions or requests).

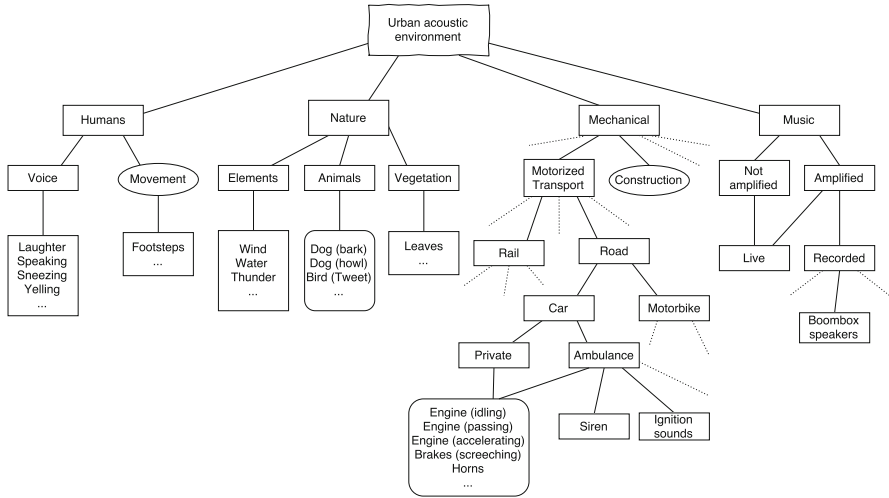


Fig. 7.8 Subset of urban sound taxonomy adapted from [52], with square boxes for sound sources, circles for actions, rounded squares for combinations of source and action

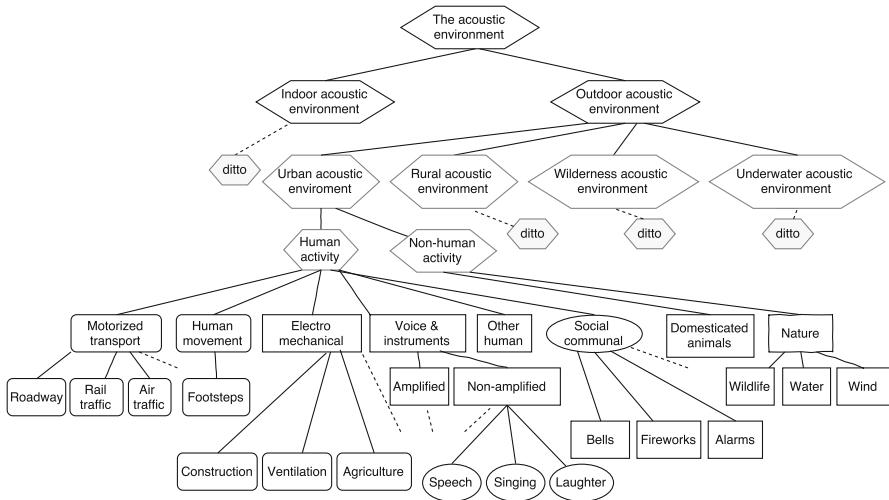


Fig. 7.9 Categorization of soundscapes adapted from [6], with square boxes for sound sources, circles for actions, hexagons for contexts

type of environment such as natural ecosystems or urban environments). From the perspective of the listeners, these could be conceived of as environments that lend themselves to particular activities, thus orienting the listening strategies to sounds that match the envisaged activity.

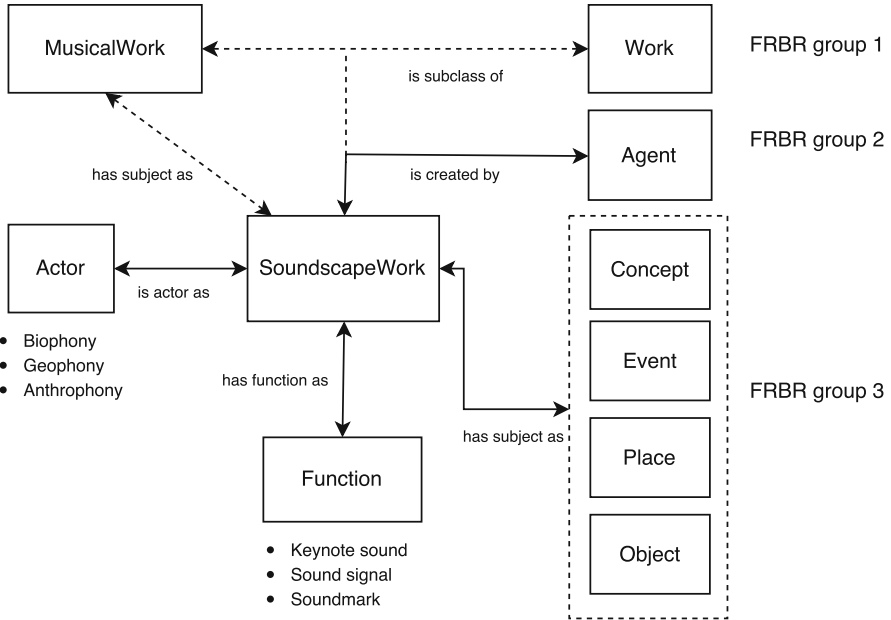


Fig. 7.10 Soundscape ontology proposed by and adapted from [8]

7.5.3 Toward a Soundscape Ontology

In [8], Choe and Ko discuss the growing use of Web technologies for online sound archives and databases as well as sound maps⁶ around the world and review the limitations of existing ontologies to model complex soundscapes. They propose an ontology for archiving soundscapes based on Functional Requirements for Bibliographic Records (FRBR) [32]. The design of this ontology, shown in Fig. 7.10, relies on a pre-existing Music Ontology as well as on Pijanowski’s classification of sound producing agents (geophony, biophony, and anthrophony, see [48]) for the entity Actor and Shafer’s classifications of different functions of sound for the entity Function.

This ontology has been developed to preserve recorded soundscapes for archival purposes. It might however be of interest to inform the development of a rich metadata scheme to describe sound events and sound scenes in the context of machine learning.

⁶For example, www.montrealsoundmap.com/.

7.5.4 *Sound Events*

Research on everyday sound perception and categorization suggests that sound events give rise to complex cognitive categories based on a correlate of attributes related to sound source(s), action(s), and context(s). We reviewed converging evidence that different principles of categorization co-exist and operate together, and that listeners are flexible in the types of categories they form and use in everyday life. As an example in the auditory domain, a dog lapping water produces a sound event that could be categorized as sound of liquid, alimentation sound, animal sound, and/or kitchen sound. Similarly, shouting can be categorized as a speech sound (e.g., cheering for a sports team) or a warning signal depending on the situation. This cross-classification into different categorization schemes allows listeners to get a full understanding of their environments in a goal-oriented view (e.g., based on the function of the sound, the identity of the sound source, to anticipate further action or inaction appropriate in a given context).

Based on the perceptual studies reviewed in Sect. 7.4, we now derive taxonomies in an attempt to represent and reconcile the different categorization schemes observed for sound events based in the literature. While these categories are not exhaustive, they represent a synthesis of what has been documented in the literature. We believe that these separate taxonomies might be useful for the design of computational systems to provide a more detailed representation of sound events in terms of a combination of sound source(s), action(s), and context(s). Such a representation could provide additional cues for computational systems to resolve ambiguity (e.g., identify context based on sound sources) and guide what could be considered an acceptable generalization or confusion (e.g., misclassifying an action but correctly identifying the source and context, or confusing the sound of a small kitchen appliance with another appliance while staying in the kitchen context).

A taxonomy of sound sources derived from previous studies is presented in Fig. 7.11. At a first level, a distinction is made between sounds produced by either animate agents or inanimate agents (e.g., [18, 22]). Animate agents then subdivide into humans and animals, while inanimate agents subdivide into different types of material (e.g., [17, 30]), and then into different types of objects. We refer the reader to research on everyday objects (e.g., Rosch's work on natural categories) for further detail on subordinate levels of categorization.

A taxonomy of actions producing sound derived in shown in Fig. 7.12. Different organizational principles have been observed for actions produced by animate and inanimate agents, a distinction also shown in terms of sources in Fig. 7.11. For animate agents, there is a first distinction between action and non-action sounds. Non-action sounds involve nothing by the body as a source, they include vocalization and body sounds, while action sounds subdivide into locomotion, alimentation, and others. At a lower level, the possible set of actions is constrained by the type of agents. As an example, under vocalization, while humans might talk, laugh, and cry, different animals might call, bark, or moo. Similarly for locomotion, while humans can walk, run, and swim, only birds can flap wings and fly. This close interaction between action and sound source is captured in Fig. 7.12 by using italics for animal-specific actions.

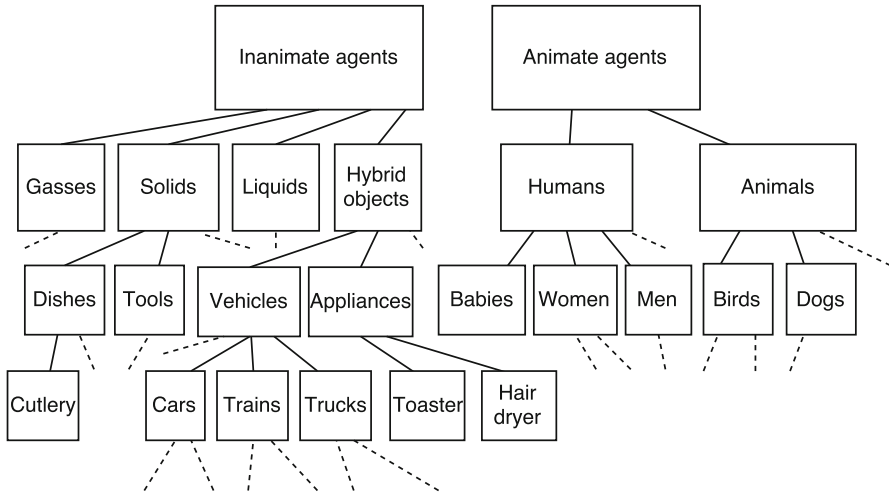


Fig. 7.11 Taxonomy of sound sources derived from a synthesis of previous studies

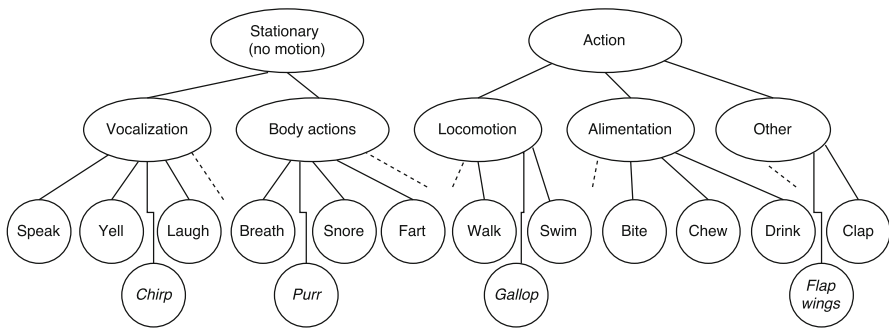


Fig. 7.12 Taxonomy of actions producing sound derived from a synthesis of previous studies. The set of actions is constrained by the type of agents (animal-specific actions shown in *italics*)

Finally a taxonomy of contexts is presented in Fig. 7.13. This form of categorization based on contexts in which everyday sounds are heard is closer to the notion of script categories that rely on routines (e.g., eating out, commuting, jogging in the park, doing the dishes). Context is particularly relevant for computational systems. Users are more likely to forgive the occasional infelicity in sound event identification as long as the broader context is preserved.

These taxonomies could be integrated to provide a rich description of sound events in terms of correlates of attributes across different descriptor layers (related to context, source, and actions) corresponding to different categorization schemes, each along different levels of abstraction, from general abstract descriptors to more specific descriptors. This is illustrated in Fig. 7.14 in the case of the sound of dog lapping water from a bowl, described in terms of context, agent, material, and action.

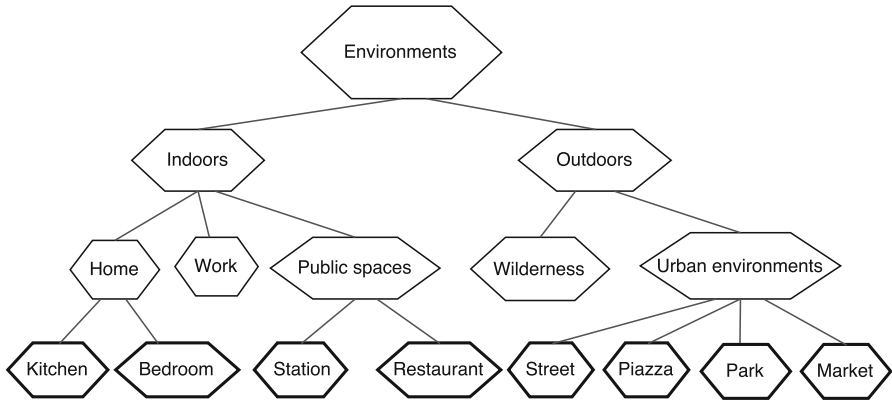


Fig. 7.13 Taxonomy of contexts derived from a synthesis of previous studies

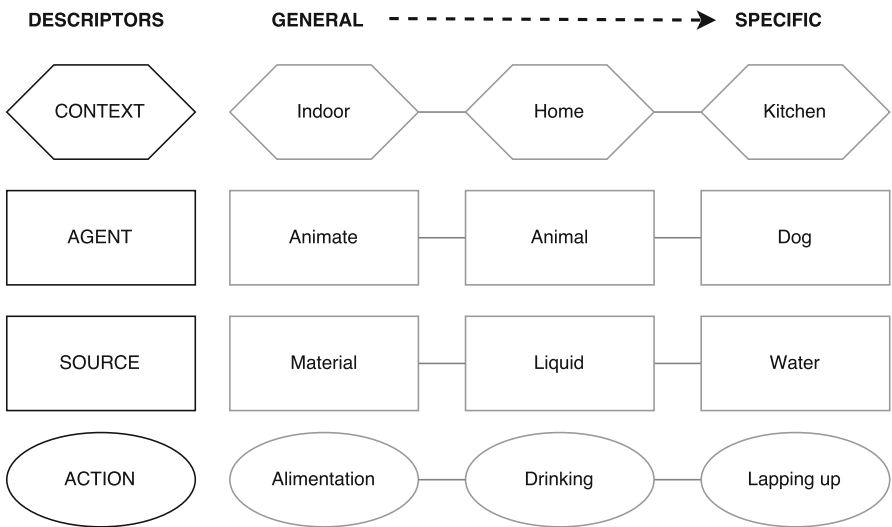


Fig. 7.14 Cross-classification of sound events into descriptors of context, agent, source, and action, illustrated here with the sound of a dog lapping water from a bowl

7.5.5 Comparison

We reviewed categorization models proposed in the literature. These are primarily based on taxonomies, that is, hierarchical classification schemes with mutually exclusive categories. This model is in the logical continuation of the rule-based view of the classical approach to categorization with strict category boundaries. However, the different levels of abstraction are compatible with Rosch’s prototype theory in terms of superordinate, basic, and subordinate levels. While the design of these taxonomies has been informed or inspired by empirical studies, the complex

cognitive strategies of human categorization cannot be entirely captured by these models. Previous research on sound events has shown that people rely on multiple categorization principles based on sound source(s), action(s), and context. We synthesized the main categories of sources, actions, and context(s) investigated in previous studies and argued that sound events should be characterized by correlates of attributes from these different categorization schemes.

7.6 Conclusion

In this chapter, we started by reviewing the historical forces that have driven categorization research and describing the most common methods for categorization research. We then summarized the main findings of previous studies on everyday sound categorization by humans to gain some perspective of what cognitive psychology has to say about the computational analysis of sound scenes and events. A first important aspect of human categorization is the existence of a basic categorization level. This basic level for everyday sounds could be appealing for the computational analysis of sound events. In computational systems, the enumeration of the leaf nodes is usually the most important set, while human categorization tends to favor a basic level, which is in the middle, with more general abstractions above it, and specializations below it. In addition this basic level is typically associated with simple words that could be used as explicit labels to supervise the training of machine learning systems.

Another specificity of human categorization discussed in this chapter is the ability to *cross-classify* into different forms of categorizations including different taxonomies (of sources, of agents, of actions, etc. . .) and script categories related to the context in which these sounds are typically heard (e.g., sounds from an outdoor market, kitchen sounds). Cross-classification implies that there is not a single way of categorizing everyday sounds. A given sound can co-exist in different categorization schemes simultaneously, each more or less appropriate in different contexts. This idea challenges the latent assumption of a unique well-defined hierarchical structure underlying many computational systems and highlights the context dependence of categorization.

Further research is needed to provide a comprehensive model of sound event categorization accounting for the different types of similarities (acoustic, causal, and semantic) as well as person-related factors (e.g., expertise, developmental stage) and situational factors (e.g., activity, context) and the interaction between these different factors. Indeed there is converging evidence that sound events give rise to complex categories relying on correlates of (non-independent) attributes related to sound source, action, and context. The relationships between these different categorization schemes could be modeled using faceted classification. Facets are mutually exclusive and jointly exhaustive categories isolating one perspective. Facets corresponding to sound sources, actions, and contexts could be combined to provide a more comprehensive description of sound events. Future research

should also investigate the heuristics governing sound scenes evaluation, including attention and memory effects (see [58]) and the relationship between isolated sound events and holistic sound scenes.

Acknowledgements Dan Ellis, Tuomas Virtanen, Mark Plumbley, Guillaume Lemaitre, Julian Rice, Christopher Trudeau and Daniel Steele for insightful comments on previous versions of this chapter.

References

1. Ballas, J.A.: Common factors in the identification of an assortment of brief everyday sounds. *J. Exp. Psychol. Hum. Percept. Perform.* **19**(2), 250–267 (1993). doi:10.1037/0096-1523.19.2.250
2. Barnes, J. (ed.): *The Complete Works of Aristotle, The Revised Oxford Translation*, vol. 1. Princeton University Press, Princeton, NJ (1984)
3. Barsalou, L.W.: Ad hoc categories. *Mem. Cogn.* **11**(3), 211–227 (1983). doi:10.3758/BF03196968. <http://link.springer.com/article/10.3758/BF03196968>
4. Berland, A., Gaillard, P., Guidetti, M., Barone, P.: Perception of everyday sounds: a developmental study of a free sorting task. *PLOS ONE* **10**(2) (2015). doi:10.1371/journal.pone.0115557. <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0115557>
5. Bregman, A.: *Auditory Scene Analysis*. MIT, Cambridge, MA (1990)
6. Brown, A.L., Kang, J., Gjestland, T.: Towards standardization in soundscape preference assessment. *Appl. Acoust.* **72**(6), 387–392 (2011). doi:10.1016/j.apacoust.2011.01.001
7. Bruner, J.: *Acts of Meaning*. Harvard University Press, Cambridge, MA (1990)
8. Choe, S.H., Ko, Y.M.: Collective archiving of soundscapes in socio-cultural context. In: *iConference 2015 Proceedings*. iSchools (2015). <https://www.ideals.illinois.edu/handle/2142/73464>
9. Clark, H.H., Wilkes-Gibbs, D.: Referring as a collaborative process. *Cognition* **22**(1), 1–39 (1986)
10. D’Andrade, R.: Some propositions about the relations between culture and human cognition. In: Stigler, J.W., Shweder, R.A., Herdt, G. (eds.) *Cultural Psychology: Essays on Comparative Human Development*, pp. 65–129. Cambridge University Press, New York, NY (1990)
11. David, S.: Représentation d’objets sensoriels et marques de la personne. In: Dubois, D. (ed.) *Catégorisation et cognition: de la perception au discours*, pp. 211–242. Kime, Paris (1997)
12. David, S., Dubois, D., Rouby, C., Schaal, B.: L’expression des odeurs en français: analyse lexicale et représentation cognitive. *Intellectia* **1**(24), 51–83 (1997)
13. Delage, B.: *Paysage sonore urbain*. Technical Report, Plan Construction, Paris (1979)
14. Dubois, D.: Categories as acts of meaning: the case of categories in olfaction and audition. *Cogn. Sci. Q.* **1**, 35–68 (2000)
15. Dubois, D., Guastavino, C., Raimbault, M.: A cognitive approach to urban soundscapes: using verbal data to access everyday life auditory categories. *Acta Acust. United Acust.* **92**(6), 865–974 (2006). http://www.academia.edu/24311328/A_cognitive_approach_to_urban_soundscapes_Using_verbal_data_to_access_everyday_life_auditory_categories
16. Ehrenfels, C.V.: On Gestalt-qualities. *Psychol. Rev.* **44**(6), 521–524 (1937). doi:10.1037/h0056968
17. Gaver, W.W.: What in the world do we hear?: an ecological approach to auditory event perception. *Ecol. Psychol.* **5**(1), 1–29 (1993). doi:10.1207/s15326969eco0501_1. http://dx.doi.org/10.1207/s15326969eco0501_1

18. Giordano, B.L., McDonnell, J., McAdams, S.: Hearing living symbols and nonliving icons: category specificities in the cognitive processing of environmental sounds. *Brain Cogn.* **73**(1), 7–19 (2010). doi:10.1016/j.bandc.2010.01.005
19. Giordano, B.L., Guastavino, C., Murphy, E., Ogg, M., Smith, B., McAdams, S.: Comparison of methods for collecting and modeling dissimilarity data: applications to complex sound stimuli. *Multivar. Behav. Res.* **46**, 779–811 (2011)
20. Goldstone, R.L.: The role of similarity in categorization: providing a groundwork. *Cognition* **52**(2), 125–157 (1994)
21. Guastavino, C.: The ideal urban soundscape: investigating the sound quality of french cities. *Acta Acust. United Acust.* **92**, 945–951 (2006)
22. Guastavino, C.: Categorization of environmental sounds. *Can. J. Exp. Psychol.* **61**(1), 54–63 (2007)
23. Guastavino, C., Cheminée, P.: A psycholinguistic approach to the ecological validity of experimental settings. *Food Qual. Prefer.* **15**, 884–886 (2003)
24. Guastavino, C., Cheminée, P.: Une approche psycholinguistique de la perception des basses fréquences : conceptualisations en langue, représentations cognitives et validité écologique. *Psychol. Fr.* **48**(4), 91–101 (2003)
25. Guastavino, C., Katz, B.F.G., Polack, J.D., Levitin, D.J., Dubois, D.: Ecological validity of soundscape reproduction. *Acta Acust. United Acust.* **91**(2), 333–341 (2005)
26. Guastavino, C., Gómez, F., Toussaint, G., Marandola, F., Gómez, E.: Measuring similarity between flamenco rhythmic patterns. *J. New Music Res.* **38**(2), 129–138 (2009)
27. Guénoche, A., Garreta, H.: Can we have confidence in a tree representation? In: Gascuel, O., Sagot, M.F. (eds.) *Computational Biology. Lecture Notes in Computer Science*, vol. 2066, pp. 45–56. Springer, Berlin, Heidelberg (2001). http://link.springer.com/chapter/10.1007/3-540-45727-5_5. doi:10.10van07/3-540-45727-5_5
28. Guyot, F., Castellengo, M., Fabre, B.: Etude de la catégorisation d'un corpus de bruits domestiques. In: Dubois, D. (ed.) *Catégorisation et cognition: de la perception au discours*, pp. 41–58. Kimé, Paris (1997)
29. Gygi, B., Kidd, G.R., Watson, C.S.: Similarity and categorization of environmental sounds. *Percept. Psychophys.* **69**(6), 839–855 (2007)
30. Houix, O., Lemaitre, G., Misdariis, N., Susini, P., Urdapilleta, I.: A lexical analysis of environmental sound categories. *J. Exp. Psychol. Appl.* **18**(1), 52–80 (2012). doi:10.1037/a0026240
31. Humphreys, G.W., Riddoch, M.J., Price, C.J.: Top-down processes in object identification: evidence from experimental psychology, neuropsychology and functional anatomy. *Philos. Trans. R. Soc. B Biol. Sci.* **352**(1358), 1275–1282 (1997). <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1692003/>
32. IFLA: Functional Requirements for Bibliographic Records (Final report). Technical Report, K.G. Saur, Munich (1998). http://archive.ifla.org/VII/s13/frbr/frbr_current_toc.htm
33. ISO 12913-1:2014 - Acoustics – Soundscape – Part 1: Definition and conceptual framework (2014). http://www.iso.org/iso/catalogue_detail.htm?csnumber=52161
34. Kruskal, J.B., Wish, M.: *Multidimensional Scaling. Quantitative Applications in the Social Sciences*. Sage, Newbury Park, CA (1978)
35. Lamberts, K., Shanks, D.: *Knowledge Concepts and Categories*. Psychology Press, East Sussex (2013)
36. Lemaitre, G., Houix, O., Misdariis, N., Susini, P.: Listener expertise and sound identification influence the categorization of environmental sounds. *J. Exp. Psychol. Appl.* **16**(1), 16–32 (2010). doi:10.1037/a0018762
37. Maffiolo, V.: *De la caractérisation sémantique et acoustique de la qualité sonore de l'environnement sonore urbain*. Ph.D. thesis, Université du Maine, Le Mans (1999)
38. Maffiolo, V., David, S., Dubois, D., Vogel, C., Castellengo, M., Polack, J.D.: Sound characterization of urban environment. In: *Proceedings of Internoise* (1997)
39. Mantel, N.: Ranking procedures for arbitrarily restricted observation. *Biometrics* **23**(1), 65–78 (1967). doi:10.2307/2528282. <http://www.jstor.org/stable/2528282>

40. Marcell, M.M., Borella, D., Greene, M., Kerr, E., Rogers, S.: Confrontation naming of environmental sounds. *J. Clin. Exp. Neuropsychol.* **22**(6), 830–864 (2000). doi:10.1076/jcen.22.6.830.949. <http://www.tandfonline.com/doi/abs/10.1076/jcen.22.6.830.949>
41. Medin, D.L., Shoben, E.J.: Context and structure in conceptual combination. *Cogn. Psychol.* **20**(2), 158–190 (1988). doi:10.1016/0010-0285(88)90018-7. <http://www.sciencedirect.com/science/article/pii/0010028588900187>
42. Morel, J., Marquis-Favre, C., Dubois, D., Pierrette, M.: Road traffic in urban areas: a perceptual and cognitive typology of pass-by noises. *Acta Acust. United Acust.* **98**(1), 166–178 (2012). doi:10.3813/AAA.918502
43. Mzali, M., Dubois, D., Polack, J.D., Létourneaux, F., Poisson, F.: Mental representation of auditory comfort inside trains: methodological and theoretical issues. In: *Proceedings of Internoise* (2001). <https://www.mysciencework.com/publication/show/1ee34f81baad63a3250ba27c89b3d81>
44. Neuhoff, J.G.: Introduction and history. In: Neuhoff, J.G. (ed.) *Ecological Psychoacoustics*, pp. 1–13. Brill, Leiden (2004)
45. Nguyen, S.P., Murphy, G.L.: An apple is more than just a fruit: cross-classification in children's concepts. *Child Dev.* **74**(6), 1783–1806 (2003). doi:10.1046/j.1467-8624.2003.00638.x. <http://dx.doi.org/10.1046/j.1467-8624.2003.00638.x>
46. Nielbo, F.L., Steele, D., Guastavino, C.: Investigating soundscape affordances through activity appropriateness. In: *Proceedings of International Congress on Acoustics* (2013)
47. Nosofsky, R.M.: Exemplar-based approach to relating categorization, identification, and recognition. In: Ashby, F. (ed.) *Multidimensional Models of Perception and Cognition*, pp. 363–393. Lawrence Erlbaum, Hillsdale, NJ (1986)
48. Pijanowski, B.C., Farina, A., Gage, S.H., Dumyahn, S.L., Krause, B.L.: What is soundscape ecology? An introduction and overview of an emerging new science. *Landsc. Ecol.* **26**(9), 1213–1232 (2011). doi:10.1007/s10980-011-9600-8. <http://link.springer.com/article/10.1007/s10980-011-9600-8>
49. Rosch, E.: Cognitive representation of semantic categories. *J. Exp. Psychol. Gen.* **104**(3), 192–233 (1975). doi:10.1037/0096-3445.104.3.192. https://www.researchgate.net/publication/232578274_Cognitive_Representation_of_Semantic_Categories
50. Rosch, E., Lloyd, B.B.: *Cognition and Categorization*, p. 47. Lawrence Erlbaum Associates, Hillsdale, NJ (1978)
51. Ross, B.H., Murphy, G.L.: Food for thought: cross-classification and category organization in a complex real-world domain. *Cogn. Psychol.* **38**(4), 495–553 (1999). doi:10.1006/cogp.1998.0712
52. Salamon, J., Jacoby, C., Bello, J.P.: A Dataset and Taxonomy for Urban Sound Research. In: *Proceedings of the 22Nd ACM International Conference on Multimedia, MM '14*, pp. 1041–1044. ACM, New York, NY (2014). doi:10.1145/2647868.2655045. <http://doi.acm.org/10.1145/2647868.2655045>
53. Sattah, S., Tversky, A.: Additive similarity trees. *Psychometrika* **42**(3), 319–345 (1977). doi:10.1007/BF02293654. <http://link.springer.com/article/10.1007/BF02293654>
54. Schafer, R.M.: *The Tuning of the World*. Alfred A. Knopf, New York (1977)
55. Schubert, E.D.: The role of auditory perception in language processing. In: Duane, D., Rawson, M. (eds.) *Reading, Perception, and Language*, pp. 97–130. York, Baltimore (1975)
56. Smith, E.E., Medin, D.L.: *Categories and Concepts*. Harvard University Press, Cambridge, MA (1981)
57. Steele, D., Bild, E., Cynthia, T., Irene, M., Cubero, J., Guastavino, C.: A comparison of soundscape evaluation methods in a large urban park in Montreal. In: *Proceedings of International Congress on Acoustics* (2016)
58. Steffens, J., Steele, D., Guastavino, C.: New insights into soundscape evaluations using the experience sampling method. In: *Proceedings of Euronoise* (2015)
59. Steffens, J., Steele, D., Guastavino, C.: Situational and person-related factors influencing momentary and retrospective soundscape evaluations in day-to-day life. *J. Acoust. Soc. Am.* **141**(3), 1414–1425 (2017)

60. Suzuki, H., Ohnishi, H., Shigemasu, K.: Goal-directed processes in similarity judgement. In: Proceedings of the 14th Annual Conference of the Cognitive Science Society, pp. 343–348 (1992)
61. VanDerveer, N.J.: Ecological acoustics: human perception of environmental sounds. Ph.D. thesis, Cornell University (1979). Unpublished Dissertation
62. Wertheimer, M.: Untersuchungen zur Lehre von der Gestalt. II. *Psychol. Forsch.* **4**(1), 301–350 (1923). doi:10.1007/BF00410640. <http://link.springer.com/article/10.1007/BF00410640>
63. Whorf, B.L.: Languages and logic. In: Carroll, J. (ed.) *Language, Thought, and Reality: Selected Papers of Benjamin Lee Whorf*, pp. 233–245. MIT, Cambridge, MA (1941)
64. Wittgenstein, L.: *Philosophical Investigations*. *Philosophische Untersuchungen*, vol. x. Macmillan, Oxford (1953)

Chapter 8

Approaches to Complex Sound Scene Analysis

Emmanouil Benetos, Dan Stowell, and Mark D. Plumbley

Abstract This chapter presents state-of-the-art research and open topics for analyzing complex sound scenes in a single microphone case. First, the concept of sound scene recognition is presented, from the perspective of different paradigms (classification, tagging, clustering, segmentation) and methods used. The core section is on sound event detection and classification, presenting various paradigms and practical considerations along with methods for monophonic and polyphonic sound event detection. The chapter will then focus on the concepts of context and “language modeling” for sound scenes, also covering the concept of relationships between sound events. Work on sound scene recognition based on event detection is also presented. Finally the chapter will summarize the topic and will provide directions for future research.

Keywords Scene analysis • Sound scene recognition • Sound event detection • Sound recognition • Acoustic language models • Audio context recognition • Hidden Markov models (HMMs) • Markov renewal process • Non-negative matrix factorization (NMF) • Feature learning • Soundscape

8.1 Introduction

The field of sound scene analysis develops computational methods for analyzing audio recordings or audio streams from various environments. Typical tasks involve sound scene recognition (identifying the environment or context of an audio recording) and sound event detection (identifying the sound sources within a recording, along with the start and end times when a sound is produced). In

E. Benetos (✉) • D. Stowell
School of Electronic Engineering and Computer Science, Queen Mary University of London,
Mile End Road, London E1 4NS, UK
e-mail: emmanouil.benetos@qmul.ac.uk; dan.stowell@qmul.ac.uk

M.D. Plumbley
Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, Surrey GU2
7XH, UK
e-mail: m.plumbley@surrey.ac.uk

realistic environments, sound scenes are inherently *complex*: there can be multiple overlapping sounds (which is often referred to as *polyphony*), either from the same or from different sound categories. The audio may also contain ambient or background sounds. The content of in-the-wild audio is also affected by weather conditions, e.g., wind or rain. Other factors that can enhance the complexity of a recording include audio recordings from portable devices, e.g., mobile phones, robot ears [49], or backpacks-on-birds [28, 65]; all the above cases add noise from the “wearer” of the device. A final level of complexity refers to the occurrence of extremely rare sound events (“black swan events”), which has applications in security/surveillance; more information on such approaches is given in Chap. 12.

In this chapter, we introduce state-of-the-art approaches and future directions for modeling and analyzing sound scenes in multisource environments. The chapter is not restricted to a single application domain: it covers both urban and nature sounds, as well as specialized applications, e.g., detection of office sounds. Our focus is on presenting published work in the field. No explicit comparisons between methods are made in terms of performance, however, useful comparisons can be inferred by inspecting results from the DCASE 2013 [66] and DCASE 2016 [70] challenges. We also draw inspiration from audio analysis approaches for modeling multisource environments in related fields, most notably speech processing and music signal processing.

We first describe in Sect. 8.2 approaches that model a sound scene as a whole. Approaches for detecting specific sound events, both in a monophonic and a polyphonic context, are presented in Sect. 8.3. Section 8.4 then bridges the two previous sections, presenting systems for detecting sound events that take context or more generally *acoustic language* information into account. Section 8.5 then presents an alternate approach for modeling sound scenes, this time as a collection of sound events. Conclusions are finally drawn in Sect. 8.6.

8.2 Sound Scene Recognition

The overall aim of sound scene recognition (also called acoustic scene classification or audio context recognition) is to characterize the acoustic environment of an audio stream by selecting one or more semantic labels for it [66]. In its basic form the task refers to classifying an input audio recording with a single label, and can be viewed as a machine learning single-label classification problem, with links to audio classification problems in other domains such as music genre classification [68] and speaker classification [45]. Another approach is to assign a set of labels to an audio recording [25]; this is typically referred to as audio tagging, and can be viewed as a form of multilabel classification. The concept of sound scene recognition can also be applied to continuous audio streams, where an audio recording is segmented into temporal regions, each corresponding to a different acoustic environment. The sound scene label can refer either to a general setting (e.g., indoors/outdoors) or to a specific context (e.g., restaurant); see Chap. 7 for more information on the taxonomy

of sound scenes. A final approach for sound scene recognition centers on audio similarity rather than explicit labelling, e.g., the work of Cauchi et al. [18] in which a bag-of-frames model was used to derive a similarity matrix between sound scenes recorded in train stations. The automatically derived similarities were compared with a similarity matrix compiled by human participants.

8.2.1 Methods

As discussed in Stowell et al. [66], there are two main strategies found in the literature for scene recognition. One is to consider the audio recording holistically and use various types of features to characterize it as a whole. The other approach is to derive an intermediate representation of sound events over time, and link the occurrence of specific sound events or atoms to specific acoustic environments; this approach will be described in detail in Sect. 8.5, with the more common feature-based approaches described in this section.

One early attempt on the problem of sound scene classification was proposed by Aucouturier and Pachet [2], who claimed that the bag-of-frames approach is sufficient for modeling urban soundscapes (while not deemed sufficient for modeling polyphonic music). Their approach used mel-frequency cepstral coefficients (MFCCs) as features, with Gaussian mixture models (GMMs) used for classification. This work was subsequently revised in [37], where it was shown that the promising results of Aucouturier and Pachet [2] likely resulted from a dataset with low within-class variability, which was not broad or natural enough to draw conclusions.

Feature-based approaches can also be divided into two main categories: approaches using hand-crafted features and approaches using feature learning. The advantage of hand-crafted features lies in incorporating expert knowledge about acoustics, sound perception, or specific attributes on the sound scenes or sound events to be recognized. Another advantage is that hand-crafted features typically result in a compact data representation that can be used as a front-end to efficient sound scene analysis approaches. However, there is much expert knowledge which is not straightforward to translate into feature engineering, and there are many domains in which we have relatively little knowledge to help us design features. The alternative is feature learning, which circumvents these problems by analyzing datasets to determine automatically what transformations should be applied to convert input into features. *Unsupervised* feature learning methods operate without any data labelling, and are often used even when labels are available, to learn generically across all classes. Such features might then be useful across many tasks. Feature learning does not require domain knowledge and so overcomes the bottleneck of expert feature engineering: it can select compact feature representations using data characteristics not apparent to a human observer, and it can be used to select high-dimensional representations with more features than are feasible to engineer manually. The primary drawback is that feature learning

Table 8.1 Feature-based sound scene recognition approaches

Hand-crafted features		Learned features	
Features	Reference	Approach	Reference
MFCCs	[2, 37]	NMF	[18]
Various low-level features	[24, 26]	Convolutional PLCA	[6]
Gammatone features	[52]	Sparse RBM	[39]
MFCCs + recurrence analysis	[59]	Spherical k-means	[60]
HOG representation	[12, 58]	NMF/convolutional NMF	[13]

requires a large sample of data from which to generalize, which may sometimes be an issue, e.g., when working with very rare sound events.

Table 8.1 lists various feature-based scene recognition approaches, indexed either by the types of features used (in the case of hand-crafted features) or by the feature learning method used (in the case of unsupervised feature learning methods). As can be seen, most approaches using hand-crafted features use low-level features, e.g., MFCCs, zero-crossing rate, spectral flux. Certain methods use descriptors adapted from image processing: in Bisot et al. [12], subband power distribution and histogram of gradients (HOG) features are extracted from log-frequency spectrograms. Likewise in Rakotomamonjy and Gasso [58] HOG features are extracted from a constant-Q transform (CQT) spectrogram. In submissions for the scene classification task of the DCASE 2013 challenge [66], the vast majority of approaches likewise used MFCCs as features, while a subset used features inspired by computational models of the human auditory system (cochleogram representations, spectrotemporal modulation features). For the more recent DCASE 2016 challenge and its sound scene classification task [70], MFCCs are no longer as dominant: a trend is observed towards time-frequency representations such as mel spectrograms or CQT spectrograms which typically use a larger number of coefficients/bins and thus have greater frequency resolution compared to MFCCs. Such higher frequency resolution is often needed in order to analyze sound scenes in multisource and noisy acoustic environments, whereas MFCCs are typically only able to provide an estimate of the global spectral shape of an audio recording.

On feature learning approaches for sound scene recognition, most approaches attempt to learn spectral or spectrotemporal representations in an unsupervised way using matrix decomposition approaches such as non-negative matrix factorization (NMF) [38]. For example, Bisot et al. [13] use CQT spectrograms as input and learn features using several variants of NMF and convolutional NMF, where audio recordings are classified using multiclass linear logistic regression. A different approach is proposed by Salamon and Bello [60], where the feature learning pipeline consists of computing mel spectrograms, followed by principal component analysis (PCA) whitening. The spherical k-means algorithm is then applied to the whitened log-mel-spectra in order to learn a feature codebook; finally, classification is performed using a random forest classifier. For the DCASE 2013 scene classification task, only one method relied on feature learning, using sparse restricted Boltzmann machines (RBMs).

For the DCASE 2016 challenge [70], several sound scene classification approaches employed feature learning as part of deep neural network architectures. In most deep learning approaches for sound scene classification in DCASE 2016 the input includes MFCCs, a mel spectrogram, or more generally a time-frequency representation. On the recognition side, most deep learning-based approaches for this challenge task used either frame-based deep neural networks (DNNs) or convolutive neural networks (CNNs). Published methods for sound scene recognition using deep learning include the work of Piczak [53], which used CNNs with mel spectrograms as input features. More information on features and feature learning for sound scene analysis can be found in Chap. 4.

8.2.2 *Practical Considerations*

There are a few practical considerations when creating a system for sound scene recognition, beyond the choice of audio features and classifier. A first aspect is whether the system only exploits a single channel or makes use of several channels. Although the use of multiple channels can be beneficial in sound scene recognition systems especially in the presence of multiple sound sources, so far in the literature the vast majority of sound scene classification systems only exploit a single audio channel, usually by converting a stereo input to mono. This is also evident from submissions to the DCASE 2016 acoustic scene classification task [70], where the input was a binaural recording: out of 49 systems, only 9 explicitly used information from both audio channels. More information on multichannel approaches for sound scene analysis can be found in Chap. 9.

Another consideration is whether a system assigns a single label to a recording/segment or if multiple labels can be assigned. Again, most approaches for sound scene recognition in the literature approach the problem as a single-class classification task. This approach has, however, practical limitations; in Eronen et al. [24] a first attempt was made to use a sound scene hierarchy, where results are presented for each scene class individually, as well as for each high-level context category. A related issue is on evaluating only on “hard” rather than fuzzy classification decisions, which has so far been used in evaluating sound scene recognition systems. Given that the vast majority of sound scene recognition systems output a non-binary representation which expresses the probability that a sound scene belongs to a certain class, the problem could be viewed in the future as a multilabel regression task with an added temporal dimension. This is also related to the concept of audio tagging [25], which can be viewed as multilabel classification and can be applied to segments of varying duration. All above approaches also assume that the scene labels are known beforehand, which is linked to the concept of closed set recognition. In realistic cases, not all scenes can be attributed one or more semantic labels (referred to as open set recognition). Recently, Battaglino et al. [4] carried out an investigation of the sound scene classification problem in an open set scenario, and proposed a classifier and evaluation methodology tailored for the open set case.

A final consideration involves the use of temporal information. Even though the label of a sound scene can be described in terms of long-term temporal context or the repetition of specific attributes, most systems in Table 8.1, as well as submissions participating in the DCASE 2013 Scene Classification task [66], adopt a frame-based approach and disregard any temporal information. Exceptions to that include approaches that learn time–frequency patches [6, 13] or extract image-based features [12, 58]. More recently, temporal information has been used for some scene classification systems submitted to the DCASE 2016 challenge [70], in particular for neural network-based systems that either learn spectrotemporal representations (i.e., CNNs) or explicitly integrate temporal information (recurrent neural networks, time delay neural networks).

8.3 Sound Event Detection and Classification

Sound event detection refers to the process of identifying segments within an audio recording, characterized by a start and end time, as well as a label referring to a specific sound class. Firstly, paradigms and techniques for sound event detection (SED) are presented. In Sect. 8.3.2, methods for monophonic SED are discussed, followed by methods for polyphonic SED in Sect. 8.3.3. Post-processing methods for SED are presented in Sect. 8.3.4, while a discussion on sound event detection and classification is made in Sect. 8.3.5.

8.3.1 Paradigms and Techniques

In previous chapters we have worked with specific conceptions of a sound “event”: often each event is defined by its onset time, offset time, and a single categorical label (see in particular Chap. 2). Sometimes, however, the data under consideration may be the simple presence/absence of an event type, with no further temporal detail. We may think of these as different paradigms or merely different output data formats—see Fig. 8.1 for an overview. In this chapter we consider various computational techniques, and as we do we will need to reflect on the different types of output that they are designed to infer.

To take a simple example, to detect events of just one “class” we might start with events annotated as a list of onset times and offset times (Fig. 8.1c or e if events may overlap). A detector could implement this by dividing the audio into small frames, and inferring whether each of those frames contained energy from an event (cf. Chap. 2). These fine-scale presence/absence decisions could then be translated into (*onset time*, *offset time*) data in a post-processing step (as we will discuss in Sect. 8.3.4). This is common, but this frame-based route implicitly neglects the possibility of overlapping events, since a presence/absence decision does not give us this information. A very different route to the same goal is to detect event onsets,

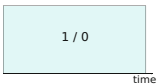
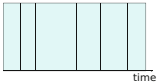
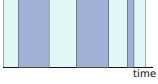
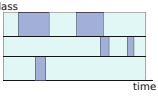
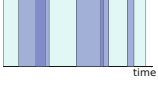
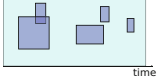
Output format		Common algorithms	Advantages	Disadvantages
(a) Presence/absence		Classifiers	Evaluation is straightforward; manual annotation can be efficient	Low temporal precision; multiple events merged
(b) Onsets		Onset detectors e.g. energy slope, per-frame classifier	Overlapping events OK; no problem with ambiguous offsets	No offset/duration information
(c) Monophonic onset-offset		Energy thresholding; HMM decoding; per-frame 0/1 classification	Many common algorithms can be used	Overlapping events merged
(d) Polyphonic onset-offset (multi-monophonic)		NMF, DNN	Joint estimation can reduce confusion between similar sound types; overlaps <i>between</i> class are OK	Cannot represent overlapping events in <i>same</i> class
(e) Polyphonic onset-offset (overlappable)			Overlapping events are OK	
(f) Time-frequency boxes		Spectrogram cross-correlation	Relates well to how human observers annotate spectrograms; indicates which frequency band relevant	Less meaningful for broadband sounds

Fig. 8.1 Output paradigms for sound event detection

and to detect event offsets, and to have some procedure for matching onsets and offsets into suitable tuples [63]. In principle this can represent overlapping events, and at any time-point the number of active events is represented, rather than “zero vs. non-zero.” However, it is less common.

Events in a sound scene have diverse characteristics, but many sounds are characterized by a relatively clear onset followed by decay to a gradual offset, the offset then being difficult for either humans or machines to localize in time. (In a few cases the onset may be ambiguous too: e.g., the sound of a car approaching and passing.) Depending on the circumstance, it may be preferable to analyze events by their onsets only, ignoring the question of when each event ends (Fig. 8.1b). This has been widely used in music signal processing [5], and in the DCASE 2013 challenge [66], where results were evaluated both for onset-and-offset and for onset-only event detection. Some methods for analyzing the events that make up a sound scene (e.g., those based on point processes) consider events as single temporal points [67].

Conversely, it may sometimes be desirable to recover even more structured information than onsets and offsets. Approaches based on template-matching of spectrogram patches naturally yield a representation of time-frequency “boxes,”

yielding events which are bounded in frequency as well as time (Fig. 8.1f). This is common in animal sound detection, because animal sounds often occupy distinct frequency ranges (see Chap. 11). If event types are bandlimited and appear in different frequency bands from one another, this can simplify the detection of different but simultaneous events.

8.3.2 *Monophonic Event Detection/Classification*

We first consider the simpler case of monophonic event detection. Monophonic here means, as in music, a situation in which only one event can be active at once (Fig. 8.1c): this is an unrealistic assumption for sound scenes in general, though it facilitates some sorts of processing, and for scenes in which sounds occur only sparsely it typically holds for most of the time. The monophonic event detection paradigm is also used in the context of detecting the predominant event in a sound scene, see, e.g., Mesaros et al. [42]. We start with the simple case with only one event type to be detected. In speech technology, voice activity detection (VAD) is an example of this scenario [10]. We will gradually move towards handling multiple event types.

In controlled environments with low noise and low amounts of clutter from unwanted event types, simple approaches can suffice. The simplest is perhaps energy thresholding: for each audio frame, the event is considered active if its energy (or power, magnitude) is above a threshold. This basic idea can be refined in various ways, such as adaptive threshold levels, or comparing the energy in different subbands of the signal. It is simple to implement and efficient, and so some variant of this is used particularly in resource-constrained detectors such as mobile phone VAD [10] or bioacoustic remote monitoring (Chap. 11), whether in itself or as a first-level step.

Another related approach is to detect event onsets, which for many event types (such as percussive sound events) can be characterized as sudden increases in energy. This approach often yields an onset-only output (Fig. 8.1b). Detecting event onsets is perhaps most well-developed in the field of music signal processing, in which it is useful for rhythm analysis and automatic transcription, and in which researchers have developed general onset detectors which are broadly applicable across a wide range of sound types: drums, string instruments, wind instruments, and so on [5]. Percussive onsets are generally the easiest to detect, due to their localized and often broadband energy.

If the events of interest are highly stereotyped, with little variation in their time and frequency characteristics (e.g., an alarm sound from an electronic device), then template matching can be a simple and robust detection method. Typically this operates by taking a time-frequency “patch” from a spectrogram as a template, and then measuring the cross-correlation between the template and the spectrogram of the signal to be analyzed. Strong peaks in the cross-correlation function are taken to

correspond to event detections [41, pp. 357–358]. These are output as lists of onsets (Fig. 8.1b) or in more detail as the bounding boxes of the time-frequency regions that matched (Fig. 8.1f).

To develop more selective event detectors, a broad strand of research uses automatic classification based on machine learning to make inferences about the presence/absence of an event in each frame (discussed in Chaps. 2 and 5) [66]. This paradigm also generalizes easily to scenarios with multiple event types of interest, simply by using a multiclass classifier rather than a binary classifier, and even to scenarios in which multiple event types can be active at once, by using a multilabel classifier. An important issue in design of such systems is not only what classifier to use, but also what feature representation to use for each audio frame. See Chap. 4 for more on this issue. Having made decisions on a per-frame basis, there is typically a need then to aggregate these into coherent event detections (see Sect. 8.3.4).

8.3.2.1 Detection Workflows

Most methods for monophonic event detection follow a process where the input audio recording is pre-processed, followed by feature extraction, recognition, and finally post-processing, as shown in Fig. 8.2. The output is typically a list of detected events, each identified by its start time, end time, and event class; or alternatively by a list of detected event classes for each time frame. Examples of pre-processing can involve audio normalization or noise reduction. In the context of event detection, feature extraction typically refers to computing a time-frequency representation of the audio signal (see Chap. 4 for more information on features). Recognition, which is typically done frame-by-frame, involves the use of a machine learning or signal processing method for detecting the presence of sound events and assigning them to a specific class (see Chap. 5 on common pattern classification methods). Finally, post-processing involves grouping instances of a recognized event class over time, in order to form a list of detected events identified with a start time, end time, and sound event class (see Sect. 8.3.4 for more details on post-processing).

In the monophonic sound event detection literature, most classification approaches are based on either probabilistic or neural network-based machine learning methods, while some are based on signal processing techniques (e.g., using hand-crafted features and rules or score functions). On feature-based methods, Plinge et al. [54] propose a bag-of-features approach using mel and gammatone frequency cepstral coefficients, with classification taking place using feature histograms.

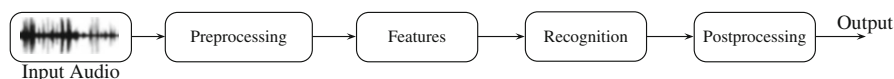


Fig. 8.2 Typical components of an event detection system

Cotton and Ellis [19] use non-negative matrix factorization (NMF) and its convolutive extension for monophonic SED; more information on NMF and spectrogram factorization approaches is given in Sect. 8.3.3. Given the temporal nature of sound event detection, several approaches use probabilistic temporal models such as hidden Markov models (HMMs), which are presented in detail in Sect. 8.3.2.2. In addition, in submissions to the DCASE 2013 Office Live task on monophonic event detection, common classifiers used include GMMs, support vector machines (SVMs), and HMMs. Phan et al. [51] address the problem of non-overlapping sound event detection using random regression forests. Finally, recent work in monophonic sound event detection involves the use of deep neural network-based approaches. These include frame-based deep neural networks [17, 23], recurrent neural networks [50], and convolutional neural networks [73]. See Chap. 5 for more details on neural network approaches for sound scene and event analysis.

8.3.2.2 Modeling Temporal Structure

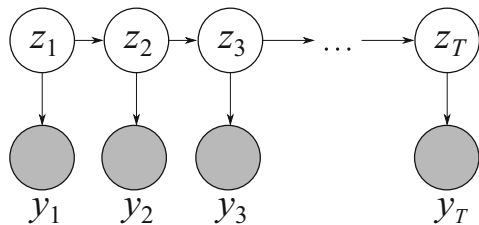
In some cases it is possible to make clear decisions based only on a single audio frame, e.g., the spectrum of a single 25 ms time window. However, in many cases sound events have temporal continuity or temporal structure which means that the immediate context before and after the current moment also supplies useful information.

A principled way to make use of such structural knowledge is to apply a probabilistic temporal model, which expresses in general terms how subsequent audio frames relate to one another [3]. Although such models may be outperformed by DNNs in large data scenarios with clear training objectives, they are useful in a wide variety of situations to perform inference given some basic prior knowledge about temporal structure.

The HMM has been widely used for this purpose. This is a model which assumes a simple form of temporal dependence: each observation depends stochastically on some unobservable “hidden state,” but only on the current value of that hidden state; and the hidden state values depend only on the immediately previous hidden state(s) (Fig. 8.3). See Chap. 5 for more details on the HMM model.

In the simplest monophonic case, the hidden state in the HMM is a binary variable indicating whether an event is currently active. To accommodate the fact

Fig. 8.3 The temporal structure expressed by a (first-order) hidden Markov model. *Arrows* represent dependence relationships. *Filled circles* represent observed variables; *empty circles*, unobserved variables



that sound events may have different characteristics as they evolve (e.g., onset vs. decay), the state space might be increased to include multiple active states for the same event, such as onset-decay-offset, with a strict left-to-right transition between these states [42]. Alternatively, or additionally, the state space might include a different “active” state for each of a set of event types of interest. This allows a single HMM to estimate for multiple event types, but please note that only one state is active at each time point, and so this is not yet a way to perform polyphonic detection.

The HMM is widely used in audio analysis, even though its temporal model is actually quite an oversimplification of most situations: it assumes that the current state, conditionally on the immediate history, is independent of all past history, which means that long-term temporal structure can be missed. A notable implication of the HMM is its effect on event durations. If the probability of exiting the present state p_{exit} is conditioned only on the state information, and not on knowledge of how long we have been in the particular state, then it must be constant for all the time we remain in the state. The prior probability of an event duration of 1 frame is thus p_{exit} , of 2 frames is $p_{\text{exit}}(1 - p_{\text{exit}})$, of 3 frames $p_{\text{exit}}(1 - p_{\text{exit}})^2$, and so on. In other words it follows a geometric distribution. A geometric distribution (or, in continuous time, an exponential distribution) is very unlike the distribution of durations we expect for most sound events. For many sound events we could express typical durations through an approximate minimum and maximum, for example, or a unimodal distribution with a nonzero mode; to build this into the model we would need something other than a geometric duration distribution. The practical consequences of this mismatch may include a tendency towards short false-positive “clutter” detections, or the inappropriate conjoining or splitting of events.

More control over durations can be achieved by augmenting the HMM with explicit probability distributions over the dwell time in each state. This is referred to as the explicit-duration HMM (EDHMM), a type of hidden semi-Markov model (HSMM) [34, 71] (Fig. 8.4).¹ In the EDHMM each step in the state sequence is associated not with one observation but a variable number of observations; upon entering state K , the number of observations to be sampled is drawn from some distribution D_K (Fig. 8.5). Figure 8.4 shows a simulation designed to illustrate conditions in which an EDHMM provides a better fit than an HMM.

The EDHMM has not often been used in event detection/classification, perhaps because of the additional implementation complexity. In some contexts there may be little gain in performance relative to a standard HMM: if the data shows strong evidence for events of specific durations, then an HMM may fit to this even if its implicit prior disfavors those durations. However, there have recently been

¹Some authors use the term HSMM to mean the same thing as EDHMM, while some use it to refer to a broader class which includes the EDHMM but also allows for further temporal structure within each state [71].

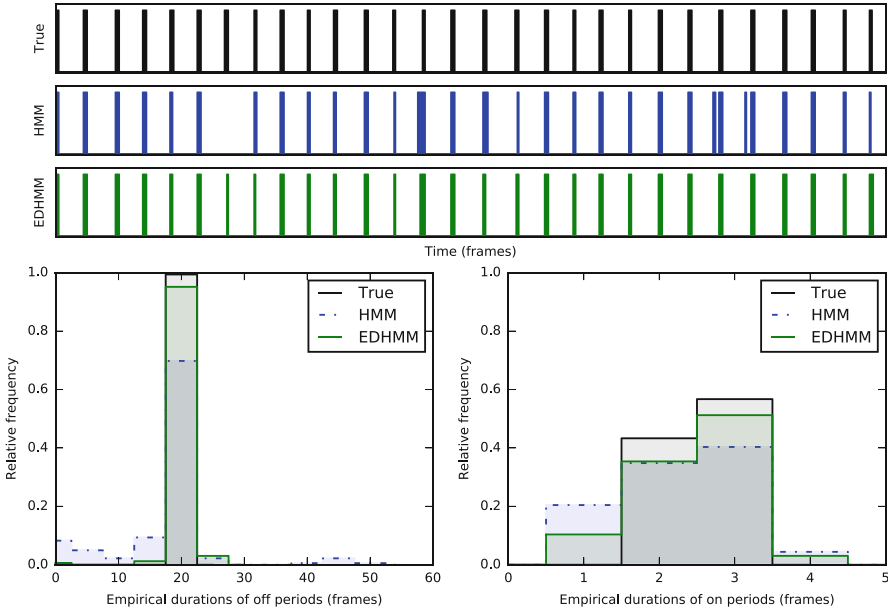
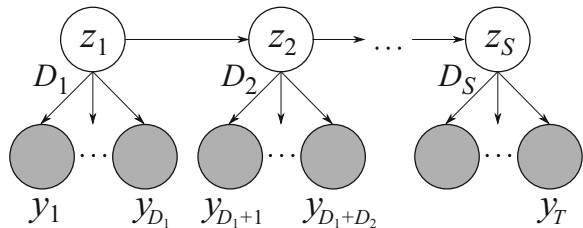


Fig. 8.4 Simulation illustrating EDHMM vs HMM. We generated spectrogram-like observations using a simple two-state model, in which the “off” state had durations drawn uniformly from the range (20,25) and the “on” state had durations drawn uniformly from the range (2,4). At each time step we sampled a observation from a Gaussian distribution whose mean depended on the current state. (The spectrogram itself is not shown, because it is very noisy—the influence of model choice is most apparent when the evidence from data is weak.) We then analyzed the observations using an EDHMM with a Poisson-distributed duration model or using an HMM. The *upper panels* show an excerpt from the timeline of on/off states, whether groundtruth (*top panel*) or inferred. (Note that neither model can perfectly recover the uniform duration distributions we used to generate the data.) Both models captured many events well, but the HMM inferred some false-positive (“clutter”) and false-negative events. This is reflected in the mismatch in empirical duration distributions of the recovered events (*lower panels*). *Implementation details:* This example was generated using the `pyhsmm` Python module, fitting a nonparametric model by Gibbs sampling [34]. We restricted the number of states to two, and for both HMM and EDHMM we ran Gibbs sampling five separate times with 150 iterations each; we plot a sample drawn from the highest-likelihood of the five runs

Fig. 8.5 The explicit-duration HMM (EDHMM), a kind of hidden semi-Markov model (HSMM) (after [34, Fig. 2]). Compare this against Fig. 8.3



methodological advances in efficient inference for EDHMMs and HSMMs [21, 34], and their use could bring advantages in disambiguating situations given uncertain evidence.

Temporal structure can be modeled in other ways than the HMM paradigm. A notable example is the approach taken by Phan et al. [51], in which a regression-forest method is used to predict from acoustic features, the time gap from each individual 100ms frame to its nearest onset and offset. These per-frame predictions are accumulated to create a kind of “heatmap” of where on the time axis we have the strongest evidence for an onset or an offset. The heatmap can then be post-processed to yield a list of events as onset-offset pairs.

One way to include temporal context is directly via the choice of features. For example, instead of taking features from the current frame only, the features from a few surrounding frames could be stacked together into a larger feature vector. In the context of deep neural networks, this idea generalizes to the structure of a CNN (also called ConvNet): each successive layer in a CNN calculates features from a local region in the data. Recurrent neural networks (RNNs) offer an alternative approach to modeling temporal context, enabling a system to make use of information extracted arbitrarily far back in time. Common to these DNN methods (see also Chap. 5) is their flexibility, but also their relative inscrutability: it is difficult to get a clear picture of the key temporal structure that a trained network has learnt, or to impose prior beliefs about that structure on a network a priori. For existing work on SED using DNNs and extensions, see [17, 23, 50, 73].

Instead of stacking multiple frames together, one could encode the amount of change in feature values from the previous frame to the current one. These “delta” features estimate the instantaneous rate of change of the feature values; likewise “delta-delta” features estimate their acceleration. Such features encode a very similar type of local temporal information as do stacked frames. They have long been used to add temporal change information into features such as MFCCs.

8.3.3 Polyphonic Sound Event Detection/Classification

So how can a system detect events while allowing for polyphony, i.e., allowing for multiple events to be simultaneously active? This issue is particularly important when analyzing dense sound scenes, or when the bias created by the monophonic constraint may not be acceptable.

8.3.3.1 Multiple Monophonic Detectors

The simplest approach is to run multiple independent monophonic detectors in parallel, perhaps one for each event type. This has practical advantages: the independent detectors can each be trained, calibrated, and modified without affecting the others, and the set of detectors to deploy can even be determined at runtime (perhaps

based on contextual information). A potential disadvantage is that, depending on the detection method, the detectors lose the advantage of shared knowledge. For example, detectors for two different but acoustically similar event types might tend to both trigger when only one event is present, rather than interacting with each other to trigger only the more likely. Cakir et al. [16] found that using a set of independent detectors worked almost as well as a multilabel detector, and thus recommended the approach on the basis of its flexibility: if the detectors are independent, then we can select at runtime which detectors are relevant for the task at hand and use a subset.

Another clear route in to polyphonic analysis is to apply a *source separation* algorithm, which takes an input signal and decomposes it as the sum of multiple component audio streams. Each of the audio streams output from the source separation process can then be subjected to event detection [30].

8.3.3.2 Joint Approaches

The above approaches are based on multiple monophonic detectors. Other recent research instead attempts to perform polyphonic detection in a single integrated system. In the following we will consider polyphonic versions of paradigms already encountered—classifiers and HMMs—as well as matrix factorization methods and deep learning approaches.

For systems based on per-frame classification, the modification is relatively simple: one can use a multilabel classifier, which is a classifier in which any number of the target classes rather than just one may be given a positive decision [16, 17]. For multilabel classification one may need to pay special attention to the choice of features used to represent the data: the best results may well be obtained using a representation in which the energies due to the different simultaneously active events do not interfere with each other, for example, if they lie on different dimensions of the feature space. This is why polyphonic sound event detection approaches use as input time-frequency representations with a high temporal and frequency resolution (see, e.g., methods participating in the DCASE 2016 challenge, Task 2 [70]). It is worth noting though that certain classifiers (e.g., neural networks) allow multilabel classification easily, whereas others (e.g., decision trees, GMMs) do not.

There are various ways that an HMM can be used to recover polyphonic sound event sequences:

- Apply a monophonic HMM in a multi-pass scheme, merging the sequences recovered from each pass [31].
- Use a single HMM in which the hidden states correspond to all event combinations, i.e., for K classes there will be 2^K hidden states. The number of states can be reduced pragmatically, by only considering the state combinations that are encountered in the training data. This is the approach used by Stowell et al. [65].
- Support multiple simultaneous streams using factorial HMMs [46, Ch. 17]. Factorial HMMs model multiple independent Markov chains under a single observation. An example use of factorial HMMs in audio source separation was proposed by Mysore et al. [48].

A separate methodological strand has focused on *matrix factorization* techniques. Such techniques treat a (non-negative) magnitude spectrogram or some other time-frequency representation as a matrix to be decomposed as a product of two lower-rank matrices. NMF decomposes the observed spectrogram/matrix \mathbf{X} with dimensions $m \times n$ as a product of two non-negative matrices: $\mathbf{X} \approx \mathbf{WH}$. Matrix \mathbf{W} has dimensions $m \times k$ and \mathbf{H} is $k \times n$ [38]. When analyzing a spectrogram with m frequency bins and n audio frames, the outcome of optimization is that \mathbf{W} becomes a matrix holding k spectral templates, and \mathbf{H} a matrix holding k time-series activation weights, specifying for each frame how the templates must be additively mixed to produce the spectrogram approximation.

NMF can be used directly for the SED problem: if we imagine that each of the k templates corresponds to a sound event type, then \mathbf{H} directly tells us the strength of presence of each event in each frame; this merely needs post-processing to recover discrete events [43, 66]. An example NMF decomposition of a sound scene is shown in Fig. 8.6; activations corresponding to the two sound event classes present in the recording are clearly visible in Fig. 8.6c. An important question is how to encourage the procedure to learn templates which correspond to individual event types. One way to do this is to initialize or even hold the templates fixed, where templates are learned in a training stage from isolated sounds [27]. Alternatively, if annotated polyphonic event sequences are available for training, then NMF can be used in a supervised fashion. In Mesaros et al. [44] this is achieved through coupled NMF: learning NMF templates by analyzing a matrix in which the spectrogram and the known activation matrix are stacked together.

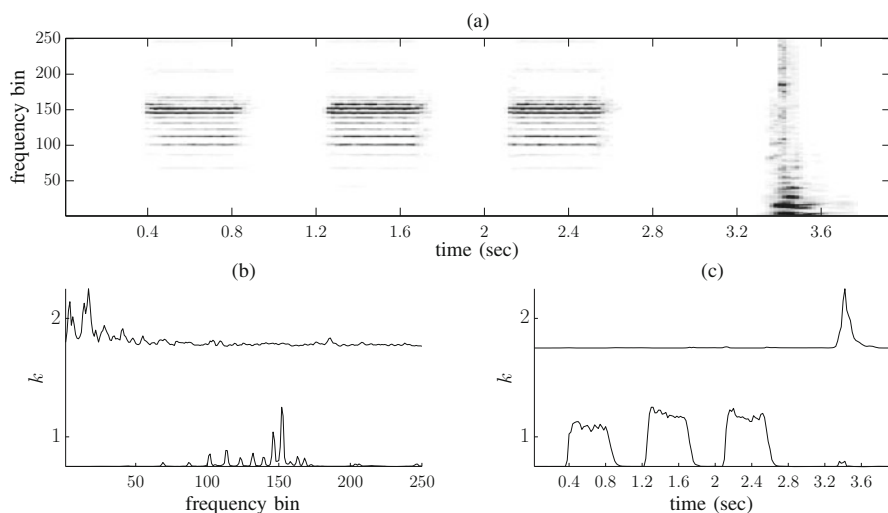


Fig. 8.6 NMF decomposition (with $k = 2$) of a sound scene consisting of three alarm sounds followed by a door slam sound. (a) The input spectrogram \mathbf{X} . (b) The spectral basis matrix \mathbf{W} . (c) The sound event activation matrix \mathbf{H}

The basic NMF model imposes no temporal structure on the problem. Various augmentations of NMF have been explored; the most significant is perhaps convolutive NMF (cNMF), in which the templates are not single spectral slices, but time-frequency patches. The signal is modeled as the additive combination of the activations \mathbf{H} convolved with their corresponding time-frequency patches.

Convolutive NMF was used in [19] for event detection in meeting room audio, and found to give robust performance. An alternative to computationally expensive cNMF for sound event detection was proposed by Gemmeke et al. [27], where time-frequency patches are stacked as vectors and standard NMF is applied.

Recent methods for polyphonic SED that are based on deep learning methods support polyphony as a multilabel classification problem, with the classifier applied for each time frame of the signal under analysis. Deep architectures build a feature hierarchy, where in each layer higher level features are extracted implicitly by the composition of lower level features [17]. In the case of DNNs, they are composed of an input layer, several hidden layers with nonlinear activation functions, and an output layer. The input vector \mathbf{x}_t contains the features corresponding to time frame t . The output vector \mathbf{h}^k (with dimensions $M^{(k)}$) for the k th layer is calculated from the weighted sum of the outputs for the previous layer \mathbf{h}^{k-1} [17]: $\mathbf{h}^k = f(\mathbf{W}^k \mathbf{h}^{k-1} + \mathbf{b}^k)$, where $f(\cdot)$ is a nonlinear activation function, \mathbf{W}^k is a weight matrix between the $(k-1)$ th and k th layers with dimensions $M^{(k-1)} \times M^{(k)}$, and \mathbf{b}^k is the bias vector for the k th layer, with dimensions $M^{(k)}$. Algorithms such as stochastic gradient descent are used for DNN learning.

Cakir et al. [17] used DNNs for polyphonic SED, where the network output is a multilabel encoding of sound events present in a time frame. This approach was extended in [50], where a multilabel bi-directional long- and short-term memory (BLSTM) RNN was used for polyphonic SED in real life recordings. Finally, in [73] CNNs were used for SED, using a de-noised spectrogram as input feature.

Regarding assumptions of polyphonic SED systems, Stowell and Clayton [63] point out that many “polyphonic” event detectors, including many of those mentioned above, contain an implicit constraint on their polyphony. Although many different event types may be active at once, often a system assumes that only one event *of each type* can occur at once. This is clearly the constraint if applying one monophonic detector for each event type, but it is also true of many multilabel classifiers and of matrix factorization methods. The restriction can be problematic in situations such as monitoring bird flocks in which many of the same call may be heard at once; Stowell and Clayton [63] describe a method based on onset- and offset-detection designed for this case. Note also that template-matching and convolutive NMF avoid this issue in that they can detect separate but overlapping events.

8.3.4 *Post-Processing*

Many of the event detection methods we have encountered produce frame-by-frame outputs of event presence/absence. For some applications this format may be useful in itself, but more commonly the desired output is a transcript of coherent event objects, specified as a list of items each having an onset time, an offset time, a label, and perhaps additionally a confidence level in the event.

Since most frame-based event detection systems produce a non-binary representation of event presence over time (posteriograms or event activations such as in Fig. 8.6c), a common post-processing approach is to perform simple thresholding on the aforementioned representation. In [17, 23] the posteriogram outputs are post-processed using a median filter prior to thresholding. Benetos et al. [7] apply minimum duration pruning after thresholding, in order to remove sound events with a small duration. In addition, Gemmeke et al. [27] used a moving average filter with an event-dependent duration.

Another class of post-processing methods involve HMM-based post-processing. For the case of monophonic event detection, [19, 27, 42] HMMs with Viterbi decoding is typically applied, where each state corresponds to a sound event class. The Viterbi decoding recovers a definite, discrete maximum-likelihood on/off sequence from the probabilistic posterior. For polyphonic sound event detection, Heittola et al. [31] use multiple restricted Viterbi passes, with the number of passes depending on the expected polyphony of the acoustic material.

An alternative post-processing approach takes place in the study by Phan et al. [51], where the confidence of onset and offset positions is computed using posterior probabilities of onset and offset displacements, in the context of an event detection system using random forests. Similar onset/offset presence posteriors are also computed in the context of bird population monitoring [63].

Beyond the field of sound scene analysis, tracking multiple concurrent sound events has been also addressed in the context of automatic music transcription. While the vast majority of transcription approaches perform filtering followed by thresholding and minimum duration pruning (e.g., [20]), an HMM-based approach was also proposed by Poliner and Ellis [55] in which the output posteriogram is binarized using class-independent 2-state on/off HMMs. While the 2-state HMMs are not able to model class interactions, they do provide temporal smoothing in the case of unconstrained polyphony.

8.3.5 *Which Comes First: Classification or Detection?*

So far we have encountered various methodologies which perform sound event detection/classification. In a working system, should the detection step come first, or the classification step, or can the steps be merged? The approaches taken differ in their answer to this question.

It may seem that detection must come before classification—after all, how can something be classified if it has not been detected? Indeed, many approaches follow this route, for example, by performing general-purpose onset detection, segmentation, or energy thresholding, and then passing the selected audio region to a classifier.

However, it is also common to use a system which makes fuzzy or frame-wise classification decisions, and then to apply a method such as thresholding or clustering to convert those intermediate decisions into detected events. This has been standard in speaker diarization, in which individual frames would be classified on a per-speaker basis, and speech regions would then be recovered through agglomerative or divisive clustering [69]. More generally, a widespread approach is to apply a Gaussian mixture model (GMM) or a DNN to each frame, which yields likelihood (or pseudo-likelihood) values, one for each class having been responsible for generating the frame; these values can be thresholded, or the maximum likelihood class selected for each frame, to convert likelihoods into an event transcript.

More recently though, the state of the art has featured single-pass methods, in which there is not a clear separation between detection and classification. In speaker diarization these are often HMM-based [1]. Since the advent of deep learning in the 2000s, neural networks have been found to give strong performance for event detection/classification, using architectures such as convolutional neural networks [17, 60] (see also Chap. 5). These neural network approaches usually perform a single integrated estimation step, although there may then be post-processing such as binarization of the output.

8.4 Context and Acoustic Language Models

8.4.1 *Context-Dependent Sound Event Detection*

Almost all sound event detection approaches presented in Sect. 8.3 assume a specific environment or context (e.g., an office environment). However, the environment of a recording can change over time, for example, when recordings are made using portable devices. So far this problem has received limited attention, although it is clear that the types of sound events directly depend on the acoustic environment. By creating an SED system with context awareness, different types of sound events can be expected or supported according to the sound scene, leading to robust detection systems. At the same time though, context recognition errors can propagate to SED errors.

An approach for utilizing context information in order to improve sound event recognition performance was proposed by Heittola et al. [31], where a context recognition step is introduced prior to event detection. The context recognition system was based on a GMM classifier using MFCCs as features, where each

individual recording is labeled with a specific context. This allows for training context-specific sound event detection systems (in this case, based on HMMs). More recently, an approach for context-based sound event detection was presented in Lu et al. [40], where a sound scene recognition and sound event recognition hierarchy is introduced; sound scenes are recognized by a 2-layer HMM and a probabilistic model is used to detect sound events given a context.

While the system of [31] is able to label each recording with a single semantic label, there is also a possibility that the context would change within a recording. A method that could be used for such cases is the switching state space model (SSSM) [46], which is essentially a hybrid discrete/continuous dynamical system. In an SSSM, assuming a sequence of observations y_t , a state s_t (referred to as the switching variable) is modeled as to take M discrete values. The observation sequence is linked with the switching variable, as well as with M sequences of latent variables $z_t^{(m)}$, $m \in \{1, \dots, M\}$, which can be either discrete or continuous. Essentially, the value of the switch variable controls the appearance of the m th latent chain. For the SED problem, s_t can represent the time-varying context label, which can activate the sound event detection model $z_t^{(m)}$ corresponding to the m th context.

Related is the hierarchical HMM [47] which could be used to model context at a higher-level component of its hidden state, and events at the lower-level component. Challenges in these methods are often in the training, which usually require more data and/or computation than standard HMMs.

8.4.2 Acoustic Language Models

So far, all models presented for sound scene recognition and sound event detection only consist of an acoustic model, and only use local temporal constraints in order to track sound events over time (e.g., frame-by-frame event transitions). This is in contrast with automatic speech recognition systems, which typically consist of an acoustic model and a language model [56]. In addition, music language models are increasingly being used in conjunction with acoustic models as part of automatic music transcription systems [14, 62]. In the context of sound scene analysis such an *acoustic language model* can be used to track sequences of events over time, predict the next occurrence and periodicity of an event, or distinguish between similar events. For example, in the context of a street sound scene, a certain periodicity of cars passing by can be observed; likewise in nature scenes a temporal regularity in bird calls is also observed.

In the context of speech processing, language modeling can refer to a statistical model [56], a grammar-based language model, or more recently to a connectionist model [72]. Statistical language models typically assign a probability to a sequence of words; a common approach for statistical language modeling is the use of n -gram models. Grammar-based models are typically based on probabilistic context-free grammars (PCFGs), which consist of production rules, each linked with a probabil-

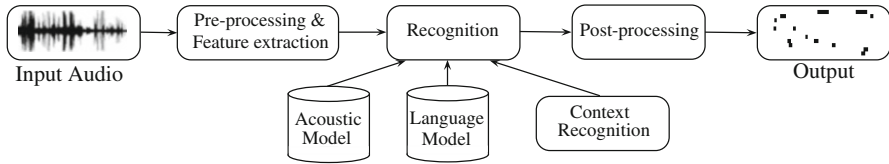


Fig. 8.7 Structure of a sound event detection system with language modeling and context recognition

ity. Neural network language models represent words as non-linear combinations of weights in a network, typically using a recurrent architecture [9].

However, unlike speech recognition, sound scenes can contain an unconstrained number of concurrent sound events, thus making common tools such as n -grams not directly applicable to the field of sound scene analysis. Recent developments, however, in music information retrieval have shown that it is possible to construct a recurrent neural network-based music language model without any constraint on the polyphony level [14], which can be combined with an acoustic model in order to create an automatic music transcription system [61]. A hierarchical approach for creating a music language model was proposed in Raczynski et al. [57], using dynamic Bayesian networks for tracking music at different levels of abstraction. To that end, a polyphonic sound event detection system can be created by integrating an acoustic model, a language model, as well as context recognition, as per Fig. 8.7.

Apart from tracking multiple concurrent sound events, acoustic language models can also be used to model the relationships between events. In Mesaros et al. [43], probabilistic latent semantic analysis was used to model the co-occurrence of annotated events in audio. Benetos et al. [8] propose a method for tracking multiple overlapping sound events using linear dynamical systems which explicitly models the co-occurrence of sound event classes. An approach for multi-object tracking applied to birdsong data was proposed by Stowell and Plumbley [64], where each event stream is modeled by a Markov renewal process. Contrary to HMMs which operate on regularly sampled time instants, a Markov renewal process (MRP) considers tuples of $\langle x_n, \tau_n \rangle$, where τ_n is the n th jump duration and x_n its associated latent state. Thus, a sequence of observations is defined as a collection of points in the time-state space instead of continuous activations (Fig. 8.8). The model thus relates very closely to a high-level model of a sound scene as an irregular sequence of event onsets (Fig. 8.1), instead of dividing each event into multiple slices at a fixed granularity. It thus offers the possibility of directly applying constraints that reflect our prior beliefs about temporal structure in the sound scene. Related to MRPs, a semi-Markov process can be defined, where a state is defined for every time instant (not just at the jump times)—in that case the end result is rather like the EDHMM discussed earlier. Note though the distinction in the observation model: under the MRP, no observations occur between events. This means that the inference procedure from data is different [64].

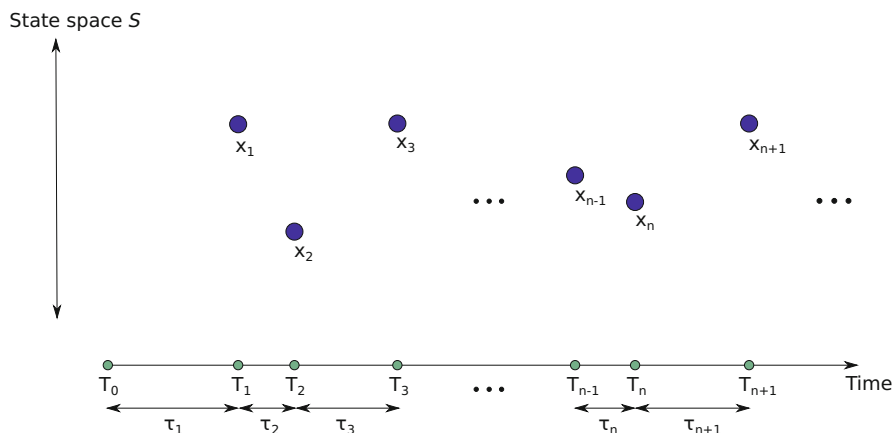


Fig. 8.8 Illustration of a Markov renewal process (MRP). This figure actually depicts any “marked” point process where each observation is characterized by its position T_n and its value $x_n \in \mathcal{S}$. This can be used as a model for the sequence of event onsets in a sound scene, where x_n indicates each event’s characteristics (such as its class or physical position). To make this a *Markov renewal process* we add the Markovian assumption that both the state x_n and the jump duration $\tau_n = T_n - T_{n-1}$ are conditionally independent of all history except for x_{n-1} . Compare against the HMM (Fig. 8.3) in which the time step is not stochastic but constant

8.5 Event Detection for Scene Analysis

We saw in Sect. 8.2.1 methods for sound scene recognition that directly model the input audio signal as a whole, using either hand-crafted features or automatically learned representations. One alternate approach to scene classification and recognition is to use information from sound event recognizers to determine the likely sound scene, based on the component events taking place in that scene.

Cai et al. [15] took this type of approach to recognize audio scenes from movies and TV shows, although using the types of *audio effects* that would be present in scenes rather than specific sound events. By detecting ten different *key audio effects* such as *laughter*, *explosion*, or *cheer*, they inferred the auditory context using a grammar network reflecting the sequences of key events together with background sounds. They compared heuristic inference, SVM-based inference, and inference based on Bayesian networks, with the Bayesian approach yielding the best results.

Heittola et al. [29] took a histogram-of-events approach to recognize real-world audio scenes such as *beach*, *bus*, or *restaurant*. HMMs were used to recognize 61 types of sound event, with the frequency of each event type counted to form an *event occurrence histogram*. Histograms were then compared using cosine difference. The authors also investigated variations including k -nearest neighbors (k -NN) classification, as well as term frequency–inverse document frequency (TF-IDF) weighting, inspired by document classification using bag-of-words models. Their best results combined their event-based recognizer with a global baseline model

using a GMM classifier of MFCC features, suggesting that the two approaches (global and event-based) bring complementary information to the scene analysis task.

Kim et al. [35] and Imoto et al. [33] introduced the idea of *acoustic topic models* to model collections of sound events (or *audio words*) that form part of a scene. Rather than modeling the entire scene as a collection of events, the events are now grouped together in *topics*, which in turn form the audio scene. For example, in their hierarchical *latent acoustic topic and event allocation (LATEA)* model, Imoto et al. [33] build a generative model of acoustic feature sequences, where the sequence of acoustic features are generated by sequences of sound events, and the sound events are in turn generated by acoustic topics. Thus a single audio scene of *cooking* may be modeled by a sequence of acoustic topics that occur during preparation of ingredients, others that occur during frying of food, and others while the food is being placed onto plates. Each of the acoustic topics can be modeled by lower-level events. This type of model can also handle situations where lower-level events are missing, using HMMs to infer their presence from surrounding events [32].

In the examples we have considered so far in this section, we think of the audio scenes as having a single label that gives its scene or context, such as “cooking” or “restaurant.” However, this is not always the case. Instead, the scene may be labeled (or *tagged*) with a collection of labels which correspond to the events within the scene, but without any indication of the time or duration of those individual events. Instead of a single label (e.g., “Changing a vehicle tire”), our audio scene may therefore be labeled with a set of labels such as {*Engine Noise, Hammering, Scraping, Clanking*}. These so-called *weak* labels provide information about what is present (or absent) in the recording, but they do not provide any more specific information such as how many events occur, when they occur, or the duration of the events [36].

Kumar and Raj [36] tackle this problem using multiple instance learning [22]. Suppose that our audio recording is composed of a sequence of short segments or frames. In a standard classifier, each segment would be labeled with positive labels for the event (or events) taking place in that segment (e.g., “This segment contains Hammering”), and, perhaps implicitly, negative labels for events which are not taking place during that segment. Instead, in multiple instance learning, we only have labels for the entire recording. All segments receive all positive and negative labels for that recording. So if a segment has the positive label “Hammering,” we know that this segment *might* contain Hammering, but we do not know for sure whether it really does. On the other hand, if a recording has a *negative* label for “Cheering,” meaning “This recording does not contain Cheering,” then we know that *none* of the component segments in that recording can contain Cheering. Hence negative labels are much stronger than positive labels.

Using this approach, Kumar and Raj [36] were able to infer events present in each segment, therefore performing event detection using only scene-level tags. Since tagging which events are present within a recording takes much less manual labor than labelling which events occur together with the time and duration of each individual event, methods like this for dealing with weak labels are likely to be important for audio scene and event recognition to deal with large scale datasets.

8.6 Conclusions and Future Directions

In this chapter, we have introduced the state-of-the-art approaches to modeling and analyzing complex sound scenes in multisource environments.

Firstly, we described sound scene recognition or audio context recognition, where the aim is to find a single label to describe the whole scene. We discussed methods that use collections of features to estimate the scene label as well as feature learning methods.

Drilling down to the constituent parts of a sound scene, we then described methods to recognize events in a sound scene. We began with the simpler case of *monophonic* event detection. For these, onset detection may be used to identify the start of events; in some cases temporal structure may be modeled, for example, using an HMM. This can be useful if the acoustic context before an event can help identify which events are more likely than others, or if the “event” itself changes over time, such as an event with a percussive onset followed by a decaying resonance.

In more complex scenes, multiple overlapping events may be present, so a single 1-of- N classification of event in each frame or segment is no longer appropriate. To handle this, several options can be used, including multiple monophonic detectors, a single classifier with multiple yes/no outputs, a classifier trained to recognize all 2^N combinations of classes, or a multilabel classifier. We also saw that a popular approach for overlapping event recognition is based on matrix factorization techniques, which decompose a time-frequency representation of the sound scene into a time-weighted sum of underlying frequency-weighted components.

We saw that events in a sound scene do not simply happen in isolation: context-dependent event detection uses the identity of the sound scene to determine which individual events are more or less likely, and hence improve event detection performance. Inspired by grammar models in speech recognition, we saw that we could also use “acoustic language models” for more general sound scenes, to represent the temporal relations between events in a sound scene.

Finally we saw how these approaches can be brought together to recognize the whole sound scene from its constituent events, either by directly recognizing the sound events as part of a scene or by introducing the idea that a sound scene is a collection of *acoustic topics*. We also saw that the scene itself may be tagged not with a single label, but instead with a collection of *weak labels* corresponding to the sound events present in the scene.

There are many avenues of future work in this field. We have encountered various methods for modeling the rich and polyphonic structure in sound scenes, from HMM-related methods (EDHMM, HHMM) and switching SSMs through to modern neural networks. Many of these methods are as yet little explored, and recent innovations in modeling and inference offer strong potential. At the time of writing, neural network methods such as RNNs are beginning to show strong performance on audio scene analysis tasks; further developments will show which architectures are best suited to this domain and can incorporate the types of prior information that are typically available.

Adaptation is an important topic deserving further attention. We have discussed the context-relativity of events; related is the fact that a trained system may be deployed in conditions which do not match those for which it was trained. The mismatch may be dramatic, or it may be a gradual “concept drift” [11]. How can a system accommodate this? Such adaptation may be automatic and continuous [11] or it could involve a small amount of bootstrap data for re-training.

Tracking “actors” in a scene through the sequence of events they emit [64] is a topic that has received little attention in the audio literature, although in the speech community it relates to speaker diarization.

One final topic to mention, relevant for practical deployment, is computation-efficient processing [62]. In most practical uses of technology, the amount of computation for a particular outcome cannot be unbounded; see Chap. 12 for more information.

The field of sound scene analysis is developing its own momentum in the specific tasks concerned and its applications. The field has connections to other existing research communities: speaker diarization, voice activity detection, robot audition, bioacoustics, music information retrieval. All these communities address problems related to the analysis of complex audio; bridging these communities brings benefit from their respective insights.

References

1. Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., Vinyals, O.: Speaker diarization: a review of recent research. *IEEE Trans. Audio Speech Lang. Process.* **20**(2), 356–370 (2012)
2. Aucouturier, J.J., Pachet, F.: The bag-of-frames approach to audio pattern recognition: a sufficient model for urban soundscapes but not for polyphonic music. *J. Acoust. Soc. Am.* **122**(2), 881–891 (2006)
3. Barber, D., Cemgil, A.T.: Graphical models for time-series. *IEEE Signal Process. Mag.* **27**(6), 18–28 (2010)
4. Battaglino, D., Lepauloux, L., Evans, N.: The open-set problem in acoustic scene classification. In: *IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)* (2016)
5. Bello, J.P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M., Sandler, M.B.: A tutorial on onset detection in music signals. *IEEE Trans. Speech Audio Process.* **13**(5), 1035–1047 (2005)
6. Benetos, E., Lagrange, M., Dixon, S.: Characterisation of acoustic scenes using a temporally-constrained shift-invariant model. In: *15th International Conference on Digital Audio Effects (DAFx)*, pp. 317–323. York, UK (2012)
7. Benetos, E., Lafay, G., Lagrange, M., Plumbley, M.: Detection of overlapping acoustic events using a temporally-constrained probabilistic model. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 6450–6454 (2016)
8. Benetos, E., Lafay, G., Lagrange, M., Plumbley, M.D.: Polyphonic sound event tracking using linear dynamical systems. *IEEE/ACM Trans. Audio Speech Lang. Process.* **25**(6), 1266–1277 (2017)
9. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. *J. Mach. Learn. Res.* **3**, 1137–1155 (2003)
10. Beritelli, F., Casale, S., Ruggeri, G., Serrano, S.: Performance evaluation and comparison of G.729/AMR/fuzzy voice activity detectors. *IEEE Signal Process. Lett.* **9**(3), 85–88 (2002)

11. Bischof, H., Godec, M., Leistner, C., Rinner, B., Starzacher, A.: Autonomous audio-supported learning of visual classifiers for traffic monitoring. *IEEE Intell. Syst.* **25**(3), 15–23 (2010)
12. Bisot, V., Essid, S., Richard, G.: HOG and subband power distribution image features for acoustic scene classification. In: 23rd European Signal Processing Conf. (EUSIPCO), pp. 719–723 (2015)
13. Bisot, V., Serizel, R., Essid, S., Richard, G.: Acoustic scene classification with matrix factorization for unsupervised feature learning. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6445–6449 (2016)
14. Boulanger-Lewandowski, N., Bengio, Y., Vincent, P.: Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In: 29th International Conference on Machine Learning, Edinburgh (2012)
15. Cai, R., Lu, L., Hanjalic, A., Zhang, H.J., Cai, L.H.: A flexible framework for key audio effects detection and auditory context inference. *IEEE Trans. Audio Speech Lang. Process.* **14**(3), 1026–1039 (2006)
16. Cakir, E., Heittola, T., Huttunen, H., Virtanen, T.: Multi-label vs. combined single-label sound event detection with deep neural networks. In: 23rd European Signal Processing Conference (EUSIPCO), pp. 2551–2555 (2015)
17. Cakir, E., Heittola, T., Huttunen, H., Virtanen, T.: Polyphonic sound event detection using multi label deep neural networks. In: International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2015). doi:10.1109/IJCNN.2015.7280624
18. Cauchi, B., Lagrange, M., Misdariis, N., Cont, A.: Saliency-based modeling of acoustic scenes using sparse non-negative matrix factorization. In: 2013 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS) (2013). doi:10.1109/WIAMIS.2013.6616131
19. Cotton, C.V., Ellis, D.P.W.: Spectral vs. spectro-temporal features for acoustic event classification. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp. 69–72 (2011)
20. Dessein, A., Cont, A., Lemaitre, G.: Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence. In: International Society for Music Information Retrieval Conference, pp. 489–494 (2010)
21. Dewar, M., Wiggins, C., Wood, F.: Inference in hidden Markov models with explicit state duration distributions. *IEEE Signal Process. Lett.* **19**(4), 235–238 (2012)
22. Dieterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.* **89**(1), 31–71 (1997)
23. Diment, A., Cakir, E., Heittola, T., Virtanen, T.: Automatic recognition of environmental sound events using all-pole group delay features. In: 2015 23rd European Signal Processing Conference (EUSIPCO), pp. 729–733 (2015)
24. Eronen, A.J., Peltonen, V.T., Tuomi, J.T., Klapuri, A.P., Fagerlund, S., Sorsa, T., Lorho, G., Huopaniemi, J.: Audio-based context recognition. *IEEE Trans. Audio Speech Lang. Process.* **14**(1), 321–329 (2006)
25. Foster, P., Sigtia, S., Krstulovic, S., Barker, J., Plumbley, M.D.: CHIME-home: a dataset for sound source recognition in a domestic environment. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (2015)
26. Geiger, J.T., Schuller, B., Rigoll, G.: Large-scale audio feature extraction and SVM for acoustic scene classification. In: 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 1–4 (2013)
27. Gemmeke, J.F., Vliegen, L., Karsmakers, P., Vanrumste, B., Van hamme, H.: An exemplar-based NMF approach to audio event detection. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (2013)
28. Gill, L.F., D’Amelio, P.B., Adreani, N.M., Sagunsky, H., Gahr, M.C., ter Maat, A.: A minimum-impact, flexible tool to study vocal communication of small animals with precise individual-level resolution. *Methods Ecol. Evol.* (2016). doi:10.1111/2041-210x.12610
29. Heittola, T., Mesaros, A., Eronen, A., Virtanen, T.: Audio context recognition using audio event histograms. In: 18th European Signal Processing Conference, pp. 1272–1276 (2010)

30. Heittola, T., Mesaros, A., Virtanen, T., Eronen, A.: Sound event detection in multisource environments using source separation. In: Workshop on Machine Listening in Multisource Environments (CHiME 2011), pp. 36–40 (2011)
31. Heittola, T., Mesaros, A., Eronen, A., Virtanen, T.: Context-dependent sound event detection. *EURASIP J. Audio Speech Music Process.* **2013**(1), 1 (2013). doi:10.1186/1687-4722-2013-1
32. Imoto, K., Ono, N.: Acoustic scene analysis from acoustic event sequence with intermittent missing event. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 156–160. IEEE, New York (2015)
33. Imoto, K., Ohishi, Y., Uematsu, H., Ohmuro, H.: Acoustic scene analysis based on latent acoustic topic and event allocation. In: 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6. IEEE, New York (2013). doi:10.1109/MLSP.2013.6661957
34. Johnson, M.J., Willsky, A.S.: Bayesian nonparametric hidden semi-Markov models. *J. Mach. Learn. Res.* **14**(Feb), 673–701 (2013)
35. Kim, S., Narayanan, S., Sundaram, S.: Acoustic topic model for audio information retrieval. In: 2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 37–40. IEEE, New York (2009)
36. Kumar, A., Raj, B.: Audio event detection using weakly labeled data. In: Proceedings of the ACM Multimedia Conference, pp. 1038–1047. ACM (2016)
37. Lagrange, M., Lafay, G., Défréville, B., Aucouturier, J.J.: The bag-of-frames approach: a not so sufficient model for urban soundscapes. *J. Acoust. Soc. Am.* **138**(5), EL487–EL492 (2015)
38. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. *Nature* **401**(6755), 788–791 (1999). doi:10.1038/44565
39. Lee, K., Hyung, Z., Nam, J.: Acoustic scene classification using sparse feature learning and event-based pooling. In: 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 1–4 (2013)
40. Lu, T., Wang, G., Su, F.: Context-based environmental audio event recognition for scene understanding. *Multimedia Systems* **21**(5), 507–524 (2015). doi:10.1007/s00530-014-0424-7
41. Marler, P.R., Slabbekoorn, H.: *Nature’s Music: The Science of Birdsong*. Academic, Cambridge (2004)
42. Mesaros, A., Heittola, T., Eronen, A., Virtanen, T.: Acoustic event detection in real life recordings. In: 18th European Signal Processing Conference, pp. 1267–1271 (2010)
43. Mesaros, A., Heittola, T., Klapuri, A.: Latent semantic analysis in sound event detection. In: 19th European Signal Processing Conference, pp. 1307–1311 (2011)
44. Mesaros, A., Heittola, T., Dikmen, O., Virtanen, T.: Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 151–155 (2015)
45. Müller, C. (ed.): *Speaker Classification I: Fundamentals, Features, and Methods*. Springer, Berlin, Heidelberg (2007). doi:10.1007/978-3-540-74200-5
46. Murphy, K.: *Machine Learning: A Probabilistic Perspective*. MIT, Cambridge, MA (2012)
47. Murphy, K.P., Paskin, M.A.: Linear-time inference in hierarchical HMMs. In: *Advances in Neural Information Processing Systems*, vol. 2, pp. 833–840 (2002)
48. Mysore, G.J., Sahani, M.: Variational inference in non-negative factorial hidden Markov models for efficient audio source separation. In: *International Conference Machine Learning (ICML)*, pp. 1887–1894 (2012)
49. Okuno, H.G., Ogata, T., Komatani, K.: Computational auditory scene analysis and its application to robot audition: five years experience. In: *International Conference on Informatics Research for Development of Knowledge Society Infrastructure (ICKS 2007)*, pp. 69–76. IEEE, New York (2007). doi:10.1109/ICKS.2007.7
50. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6440–6444 (2016)

51. Phan, H., Maasz, M., Mazur, R., Mertins, A.: Random regression forests for acoustic event detection and classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(1), 20–31 (2015)
52. Phan, H., Hertel, L., Maass, M., Koch, P., Mertins, A.: Label tree embeddings for acoustic scene classification. In: *Proceedings of the ACM Multimedia Conference* (2016)
53. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: *International Workshop on Machine Learning for Signal Processing (MLSP)* (2015). doi:10.1109/MLSP.2015.7324337
54. Plinge, A., Grzeszick, R., Fink, G.A.: A bag-of-features approach to acoustic event detection. In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3704–3708 (2014)
55. Poliner, G., Ellis, D.: A discriminative model for polyphonic piano transcription. *EURASIP J. Adv. Signal Process.* (8), 154–162 (2007). doi:10.1155/2007/48317
56. Rabiner, L., Juang, B.H.: *Fundamentals of Speech Recognition*. Prentice-Hall, Upper Saddle River, NJ (1993)
57. Raczynski, S., Vincent, E., Sagayama, S.: Dynamic Bayesian networks for symbolic polyphonic pitch modeling. *IEEE Trans. Audio Speech Lang. Process.* **21**(9), 1830–1840 (2013)
58. Rakotomamonjy, A., Gasso, G.: Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(1), 142–153 (2015)
59. Roma, G., Nogueira, W., Herrera, P.: Recurrence quantification analysis features for environmental sound recognition. In: *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (2013). doi:10.1109/WASPAA.2013.6701890
60. Salamon, J., Bello, J.P.: Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process. Lett.* **24**(3), 279–283 (2017)
61. Sigtia, S., Benetos, E., Dixon, S.: An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(5), 927–939 (2016)
62. Sigtia, S., Stark, A.M., Krstulovic, S., Plumbley, M.D.: Automatic environmental sound recognition: performance versus computational cost. *IEEE/ACM Trans. Audio Speech Lang. Process.* (2016)
63. Stowell, D., Clayton, D.: Acoustic event detection for multiple overlapping similar sources. In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (2015)
64. Stowell, D., Plumbley, M.D.: Segregating event streams and noise with a Markov renewal process model. *J. Mach. Learn. Res.* **14**, 2213–2238 (2013)
65. Stowell, D., Benetos, E., Gill, L.F.: On-bird sound recordings: automatic acoustic recognition of activities and contexts. *IEEE/ACM Trans. Audio Speech Lang. Process.* **25**(6), 1193–1206 (2017)
66. Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.D.: Detection and classification of acoustic scenes and events. *IEEE Trans. Multimedia* **17**(10), 1733–1746 (2015)
67. Stowell, D., Gill, L.F., Clayton, D.: Detailed temporal structure of communication networks in groups of songbirds. *J. R. Soc. Interface* **13**(119) (2016). doi:10.1098/rsif.2016.0296
68. Sturm, B.L.: A survey of evaluation in music genre recognition. In: *10th International Workshop on Adaptive Multimedia Retrieval: Semantics, Context, and Adaptation (AMR 2012)*, Revised Selected Papers, pp. 29–66. Springer International Publishing, Berlin (2014). doi:10.1007/978-3-319-12093-5_2
69. Tranter, S.E., Reynolds, D.A.: An overview of automatic speaker diarization systems. *IEEE Trans. Audio Speech Lang. Process.* **14**(5), 1557–1565 (2006)
70. Virtanen, T., Mesaros, A., Heittola, T., Plumbley, M., Foster, P., Benetos, E., Lagrange, M.: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*. Tampere University of Technology. Department of Signal Processing (2016). <http://www.cs.tut.fi/sgn/arg/dcase2016/>

71. Yu, S.Z.: Hidden semi-Markov models. *Artif. Intell.* **174**(2), 215–243 (2010). doi:10.1016/j.artint.2009.11.011
72. Yu, D., Deng, L.: *Automatic Speech Recognition: A Deep Learning Approach*. Springer, London (2015). doi:10.1007/978-1-4471-5779-3
73. Zhang, H., McLoughlin, I., Song, Y.: Robust sound event recognition using convolutional neural networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 559–563 (2015)

Chapter 9

Multiview Approaches to Event Detection and Scene Analysis

Slim Essid, Sanjeel Parekh, Ngoc Q.K. Duong, Romain Serizel,
Alexey Ozerov, Fabio Antonacci, and Augusto Sarti

Abstract This chapter addresses sound scene and event classification in *multiview* settings, that is, settings where the observations are obtained from multiple sensors, each sensor contributing a particular *view* of the data (e.g., audio microphones, video cameras, etc.). We briefly introduce some of the techniques that can be exploited to effectively combine the data conveyed by the different views under analysis for a better interpretation. We first provide a high-level presentation of generic methods that are particularly relevant in the context of multiview and multimodal sound scene analysis. Then, we more specifically present a selection of techniques used for audiovisual event detection and microphone array-based scene analysis.

Keywords Multimodal scene analysis • Multiview scene analysis • Multichannel audio • Joint audiovisual scene analysis • Representation learning • Data fusion • Matrix factorization • Tensor factorization • Audio source localization and tracking • Audio source separation • Beamforming • Multichannel Wiener filtering

S. Essid (✉)

LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France
e-mail: slim.essid@telecom-paristech.fr

S. Parekh • Ngoc Q.K. Duong • A. Ozerov
Technicolor, Rennes, France

e-mail: Sanjeel.Parekh@technicolor.com; Quang-Khanh-Ngoc.Duong@technicolor.com;
Alexey.Ozerov@technicolor.com

R. Serizel

Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, France
e-mail: romain.serizel@loria.fr

F. Antonacci • A. Sarti

Politecnico di Milano, Milano, Italy

e-mail: fabio.antonacci@polimi.it; augusto.sarti@polimi.it

9.1 Introduction

We now extend the study of sound scene and event classification to settings where the observations are obtained from multiple sensors, which we refer to as *multiview data*, each sensor contributing a particular *view* of the data. Instances of this include both *multichannel* audio data, as acquired by microphone arrays, or more generally, *multimodal* data, i.e., heterogeneous data that involves two or more *modalities* such as the *audio* or *visual* modalities in video recordings.

Be it for applications in machine perception—at the heart of robots’ and virtual agents’ intelligence systems—or video description—as part of video surveillance or multimedia indexing systems—multiview approaches can lead to a significant boost in performance in challenging real-world situations. Indeed, multiplying the sources of information, through different views, should result in a more robust overall “picture” of the scene being analyzed, where sensors, and consequently views, which are not reliable, e.g., noisy, at a particular time instant, are hopefully backed-up by others. This is, for instance, the case in video recordings where sound-emitting target events are not visible onscreen because of poor lighting conditions or occlusions.

Such an endeavor is actually as promising as challenging, primarily because of the significant increase in the volume of the data to be analyzed, but also owing to the potential heterogeneity of the different streams of information (e.g., audio and visual streams), which additionally may not be perfectly synchronized. Another difficulty is that it is usually not possible to determine which streams are not reliable at every time instant. To see this, consider the scenario of scene analysis using a robot’s sensors. The data views available may then be composed of the multiple audio streams acquired by the robot’s microphone array, as well as RGB and depth-image streams captured by its cameras, possibly along with other signals recorded by inertial measurement units. As the cameras are pointed at an interactant, events of interest may appear only partially in their field of view, and be present in the audio recording only at a very low signal-to-noise ratio. This may be due to background noise (including the robot’s internal noise, the so-called *ego-noise*, typically produced by its cooling fans or its actuators) and the voice of the interactant, or the robot itself, in the foreground.

In this chapter, we briefly introduce some of the techniques that can be exploited to effectively combine the data conveyed by the different views under analysis for a better interpretation. Numerous good surveys have been written on the general topic of multimodal data fusion, notably the paper by Atrey et al. [10] which is quite comprehensive. Therefore, we first provide a high-level presentation of generic methods that are particularly relevant in the context of multiview and multimodal sound scene analysis (Sect. 9.3). It is worth noting that some of the techniques presented have not necessarily yet been considered in the context of scene and event recognition as envisaged in this book. We still briefly cover them in this chapter as it is believed they hold a good potential for such applications. We then more specifically present a selection of techniques used for audiovisual event detection and microphone array-based scene analysis (in Sects. 9.4 and 9.5, respectively).

9.2 Background and Overview

9.2.1 Multiview Architectures

Figure 9.1 depicts an overview of the main fusion strategies that are exploited when analyzing multiview data, namely (1) fusion at the representation or feature level (upper row of the figure), and (2) fusion at the decision-level, usually implying integration of partial classifier-outputs. Each of these methods will be discussed further in Sect. 9.3. In particular, we will focus on a special case of representation/feature-level fusion that is here referred to as *joint subspace learning* where the aim is to learn or non-trivially transform the representations based on inter-relationships across the views.

As previously mentioned, views can be either of the same nature, in which case they are referred to as channels (typically audio channels) each corresponding to a particular microphone, or of different nature as in multimodal scenarios where, for example, some of the views could correspond to different audio channels while others to video images recorded by different cameras.

9.2.2 Visual Features

Since videos are central to the content of this chapter, a short note on commonly employed visual features is in order. Features extracted from visual streams can

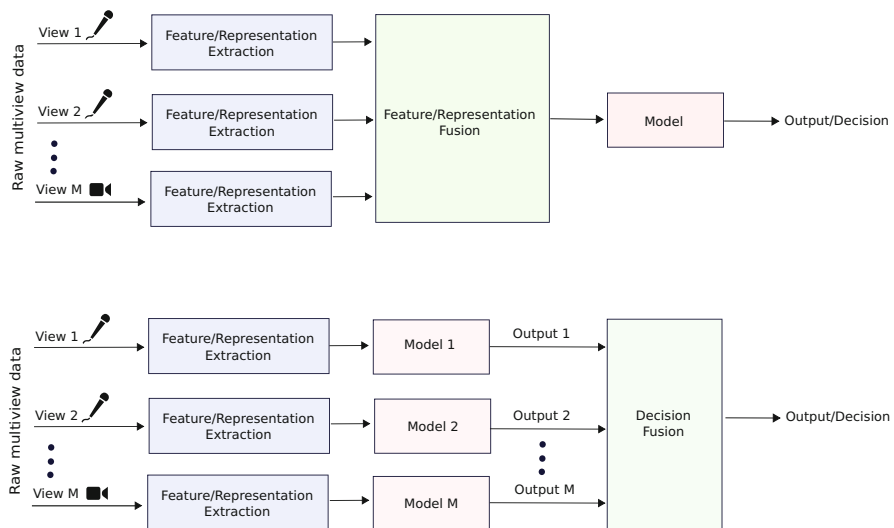


Fig. 9.1 Overview of multiview data analysis approaches

be broadly classified into two categories: appearance-based and motion-based features. For the former, several local and global descriptors representing appearance attributes namely, color, texture, and shape are extracted. While some works utilize the raw pixel data or color histograms, others rely on standard features such as scale-invariant feature transform (SIFT) [96] and histograms of oriented gradients (HOG) [40]. Lately, features extracted from convolutional neural networks have dominated [88].

Motion-based features are typically computed using optical flow or tracking data. It is possible to represent temporal changes of segmented regions, objects, and shapes by calculating velocity and acceleration, i.e., optical flow and its derivative. Other popular features include histograms of optical flow (HOF) [154] and motion boundary histograms (MBH) [154]. As MBH is computed from optical flow derivatives, it is not affected by constant motion. This makes it robust to camera motion. The reader is referred to [79, 101] for an extensive review of visual representations used for multimodal analysis.

In multiview settings temporal synchronization across views is quite challenging. Notably, in the audiovisual case, since the video frame rate, typically around 25–30 frames per second is significantly different from the audio one, features from both modalities must be appropriately sampled for temporal correspondence. Moreover, the natural asynchrony that exists between the two modalities must also be taken into account. This means that cues for an audiovisual event might not appear simultaneously in both modalities.

9.3 General Techniques for Multiview Data Analysis

Generally, the techniques discussed here (in the following two subsections) operate at either the *representation-level* or the *decision-level* as further described in the next sections.

9.3.1 Representation and Feature Integration/Fusion

Representation or feature integration/fusion is the process of combining different types of features or low-level data representations from different views into a common representation (usually to be exploited by a prediction system).

In practice, this can be primarily achieved by concatenating the feature vectors $\mathbf{o}_{m,t}$, extracted from views m , $1 \leq m \leq M$, at the same time positions t , to build integrated feature vectors $\bar{\mathbf{o}}_t = [\mathbf{o}_{1,t}^\top, \dots, \mathbf{o}_{M,t}^\top]^\top$; provided that the data analysis-rate and cross-view synchronization issues have been previously addressed.

However, the dimensionality of the resulting representation is often too high, which has led researchers to resort to dimensionality reduction methods. A common

approach is then to use *feature transform* techniques, possibly principal component analysis (PCA) [31], independent component analysis (ICA) [139], or linear discriminant analysis (LDA) [31] (see Chap. 4, Sect. 4.5.1). An interesting alternative is *feature selection* (see Chap. 4, Sect. 4.5.2). In fact, when applied to the feature vectors $\bar{\mathbf{o}}_t$, the selection will hopefully retain a subset of the most “relevant” features across the various views (with respect to a selection criterion).

Nevertheless, in multimodal settings, the previous methods often turn out to be limited owing to the different physical nature of the features to be combined. In particular, the features do not necessarily live in the same metric spaces, and are not necessarily extracted from the same temporal segments. Consequently, there has been a number of works attempting to address these limitations.

An interesting approach, within the framework of multiple kernel learning, consists in considering separate kernels for different features, to build optimal convex combinations of these in order to use them for classification, as done, for example, in [30, 157].

Another approach that is worthy of note is the construction of joint multimodal representations, as done in video analysis applications, where various types of audiovisual representations have been envisaged. Examples include the creation of *audiovisual atoms* [78] or *audiovisual grouplets* [76], both exploiting audiovisual correlations. A joint audiovisual representation may in particular be built using one of the *joint subspace learning* methods described in the following.

9.3.1.1 Feature-Space Transformation

A number of techniques have been suggested to map the observed feature vectors from two modalities to a low dimensional space where a *measure of “dependency”* between them can be computed. Let us assume the N observed feature vectors from two modalities, $\mathbf{o}_{1,t} \in \mathbb{R}^{J_1}$ and $\mathbf{o}_{2,t} \in \mathbb{R}^{J_2}$ ($t = 1, \dots, N$), are assembled column-wise in matrices $\mathbf{O}_1 \in \mathbb{R}^{J_1 \times N}$ and $\mathbf{O}_2 \in \mathbb{R}^{J_2 \times N}$, respectively.¹ The methods we describe here aim to find two mappings f_1 and f_2 (that reduce the dimensions of feature vectors in each modality), such that a dependency measure $S_{12}(f_1(\mathbf{O}_1), f_2(\mathbf{O}_2))$ is maximized. Various approaches can be described using this same formalism. The advantages of doing so are twofold: (1) it appropriately modifies the feature spaces to uncover relationships between views specified by the measure of dependency, and (2) by projecting data into the same space, dimensionality difference between views is eliminated and direct comparison across views is made possible. Fisher et al. [51] choose the mutual information [35] as a dependency measure and seek single-layer perceptrons f_1 and f_2 projecting the audiovisual feature vectors to a 2-dimensional space. Other more popular approaches (for which closed-form solutions can be found) use linear mappings to project the feature streams:

¹The underlying assumption is that the (synchronized) features from both modalities are extracted at the same rate. In the case of audio and visual modalities this is often obtained by down-sampling the audio features or up-sampling the video features, or by using temporal integration techniques [80].

- Canonical correlation analysis (CCA), first introduced by Hotelling [67], aims at finding pairs of unit-norm vectors \mathbf{t}_1 and \mathbf{t}_2 such that

$$(\mathbf{t}_1, \mathbf{t}_2) = \arg \max_{(\mathbf{t}_1, \mathbf{t}_2) \in \mathbb{R}^{J_1} \times \mathbb{R}^{J_2}} \text{corr}(\mathbf{t}_1^\top \mathbf{O}_1, \mathbf{t}_2^\top \mathbf{O}_2) \quad (9.1)$$

CCA can be considered equivalent to mutual information maximization for the particular case where the underlying distributions are elliptically symmetric [83]. Several variants have been proposed to incorporate sparsity and non-negativity into the optimization problem to resolve issues with interpretability and ill-posedness, respectively [84, 138]. In the context of multimodal neuronal data analysis, temporal kernel CCA [15] has been proposed to take into account the temporal dynamics.

- An alternative to the previous methods (expected to be more robust than CCA) is co-inertia analysis (CoIA). It consists in maximizing the covariance between the projected audio and visual features:

$$(\mathbf{t}_1, \mathbf{t}_2) = \arg \max_{(\mathbf{t}_1, \mathbf{t}_2) \in \mathbb{R}^{J_1} \times \mathbb{R}^{J_2}} \text{cov}(\mathbf{t}_1^\top \mathbf{O}_1, \mathbf{t}_2^\top \mathbf{O}_2) \quad (9.2)$$

A possible reason for CoIA's stability is that it is a trade-off between CCA and PCA, thus it benefits from advantages of both [21].

- Yet another configuration known as cross-modal factor analysis (CFA), and found to be more robust than CCA in [92], seeks two matrices \mathbf{T}_1 and \mathbf{T}_2 , such that

$$(\mathbf{T}_1, \mathbf{T}_2) = \arg \max_{(\mathbf{T}_1, \mathbf{T}_2)} (1 - \|\mathbf{T}_1 \mathbf{O}_1 - \mathbf{T}_2 \mathbf{O}_2\|_F^2) = \arg \min_{(\mathbf{T}_1, \mathbf{T}_2)} \|\mathbf{T}_1 \mathbf{O}_1 - \mathbf{T}_2 \mathbf{O}_2\|_F^2 \quad (9.3)$$

with $\mathbf{T}_1 \mathbf{T}_1^\top = \mathbf{I}$ and $\mathbf{T}_2 \mathbf{T}_2^\top = \mathbf{I}$. $\|\mathbf{V}\|_F$ denotes the Frobenius norm of matrix \mathbf{V} .

Note that all the previous techniques can be kernelized to study nonlinear coupling between the modalities considered (see, for instance, [64, 90]).

The interested reader is referred to [64, 67, 92] for further details on these techniques, and to [58] for a comparative study.

9.3.1.2 Multimodal Dictionary Learning

While previous approaches relied on modeling the association between the features across modalities, this class of techniques targets the extraction of meaningful multimodal structures to jointly represent all the modalities. This is useful because feature transformation techniques like CCA impose simplifying assumptions such as linearity and are adversely affected by lack of data. To this end, Monaci et al. [106] propose to learn multimodal dictionaries wherein the dictionary elements are learned using an algorithm that enforces synchrony between modalities and decorrelation between the learned dictionary elements. The learned templates can

then be used for performing various tasks. Monaci et al. improve upon this foundational work by proposing a bimodal matching pursuit algorithm which integrates dictionary learning and coding [107]. The sparse shift-invariant generative model used for the audiovisual case can be given by defining multimodal dictionary elements $\{\phi_d\}_{d=1}^D = (\phi_d^a(t), \phi_d^v(x, y, t))$ consisting of audio, ϕ_d^a , and visual, ϕ_d^v , parts and a spatio-temporal shift operator $T_{(pqr)}\phi_d = (\phi_d^a(t-r), \phi_d^v(x-p, y-q, t-r))$ such that the multimodal signal s is approximated by the following equation:

$$s \approx \sum_{d=1}^D \sum_{i=1}^{n_d} c_{d_i} T_{(pqr)d_i} \phi_d \quad (9.4)$$

where n_d is the number of instances of ϕ_d and c_{d_i} specifies the weights for AV components of ϕ_d at the i th instance. Several limitations of this approach have been improved upon by proposing a new objective function and algorithm to balance the two modalities, reduce computational complexity, and improve robustness [94].

9.3.1.3 Co-Factorization Techniques

Matrix factorization techniques can be profitably used to extract meaningful representations for the data being analyzed.

When dealing with *multichannel* data—i.e., with data views of the same nature (e.g., multichannel audio or images)—observations from multiple channels may be profitably assembled in *multi-way arrays*, i.e., *tensors*, before being modeled by tensor factorization methods. As for multichannel audio data, a popular approach consists in collecting the spectrograms of signals from different channels (originating from different microphones) in a 3-way tensor, as illustrated in Fig. 9.2, before processing it with the so-called PARAFAC (PARALLEL FACtor analysis) decomposition method, possibly with non-negativity constraints. This can be interpreted as an attempt to explain audio spectra observations \mathbf{v}_{tm} as being linear combinations of elementary spectra \mathbf{w}_k , temporally weighted by activation coefficients o_{kt} up to spatial modulation coefficients q_{mk} .

Such decompositions were found particularly useful in multichannel audio source separation [52, 118]. For more information about tensor factorization methods, we refer the reader to [33, 87, 159].

In contrast to the previous setting, data from different modalities usually live in feature spaces of completely different topology and dimensionality (think of audio as opposed to video), preventing the possibility of “naturally” representing them by the same tensor. In this case, one may resort to the so-called *co-factorization* techniques, that is techniques performing two (or more) factorizations in parallel, which are linked in a particular way. Because of the different nature of the modalities, this link has usually to be characterized through temporal dependencies between the temporal activations in cross-modal correspondence, and unlikely through dependencies between dictionary elements of different modalities.

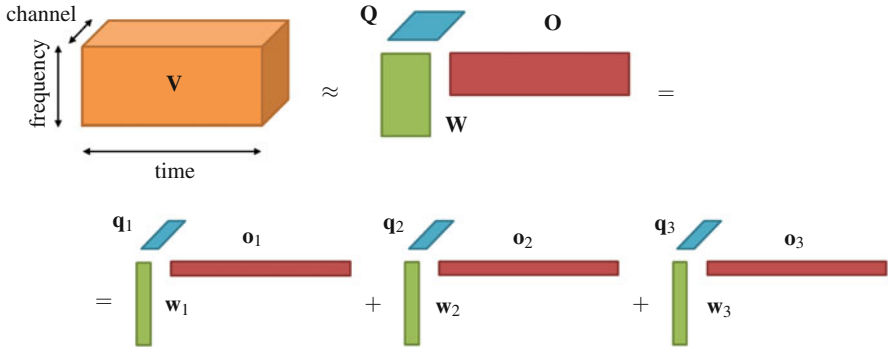


Fig. 9.2 PARAFAC decomposition of multichannel audio spectra

Assuming that appropriate nonnegative features have been extracted at the same rate from the two modalities being analyzed²—say the audio and images of a video—so that two observation matrices $V_1 \in \mathbb{R}_+^{J_1 \times N}$ and $V_2 \in \mathbb{R}_+^{J_2 \times N}$ are available, for the audio and visual data. One may seek a model (W_1, W_2, O) such that:

$$\begin{cases} V_1 \approx W_1 O \\ V_2 \approx W_2 O \\ W_1 \geq 0, W_2 \geq 0, O \geq 0; \end{cases} \quad (9.5)$$

in such a way that the temporal activations be the same for both modalities. This is referred to as *hard co-factorization*, an approach that has been followed in a number of works (see, e.g., [53, 160, 161]). Clearly, this approach is limited in that it does not account for possible local discrepancies across the modalities. This happens, for example, when there is a mismatch between the audio and the images information, say because of a visual occlusion in video analysis scenarios. This motivates the *soft co-factorization* model of Seichepine et al. [134], which merely encourages the temporal activations corresponding to each modality to be close, as opposed to equal, according to:

$$\begin{cases} V_1 \approx W_1 O_1 \\ V_2 \approx W_2 O_2 \\ O_1 \approx O_2 \\ W_1 \geq 0, W_2 \geq 0, O_1 \geq 0, O_2 \geq 0. \end{cases} \quad (9.6)$$

²To simplify, we consider the case of two modalities, but clearly the methods described here can be straightforwardly generalized to more than two data views by considering the relevant pairwise associations.

The model (9.6) is estimated by solving the following optimization problem:

$$\begin{cases} \min_{\theta} C_c(\theta) ; \theta \triangleq (\mathbf{W}_1, \mathbf{O}_1, \mathbf{W}_2, \mathbf{O}_2) \\ \mathbf{W}_1 \geq 0, \mathbf{W}_2 \geq 0, \mathbf{O}_1 \geq 0, \mathbf{O}_2 \geq 0 ; \end{cases} \quad (9.7)$$

$$C_c(\theta) \triangleq D_1(\mathbf{V}_1 | \mathbf{W}_1 \mathbf{O}_1) + \gamma D_2(\mathbf{V}_2 | \mathbf{W}_2 \mathbf{O}_2) + \delta P(\mathbf{O}_1, \mathbf{O}_2) ; \quad (9.8)$$

where:

- $D_1(\cdot | \cdot)$ and $D_2(\cdot | \cdot)$ are the measures of fit, respectively, relating to the first and second modalities; note that they may be chosen to be different divergences, each well suited to the corresponding feature space;
- $P(\cdot, \cdot)$ is a penalty on the difference between (properly rescaled) activation values occurring at the same instant; they can be either the ℓ_1 or ℓ_2 -norm of the difference between the rescaled activations;
- γ and δ are regularization parameters controlling, respectively, the relative importance of each modality and the coupling penalty.

The interested reader is referred to [134] for more details on the algorithms.³ The soft co-factorization scheme has proven effective for multichannel [134] and multimodal audio source separation [120], as well as multimodal speaker diarization [133]. It is believed to be promising for audiovisual event detection tasks.

9.3.1.4 Neural Networks and Deep Learning

Lately, rapid progress in the application of deep learning methods to representation learning has motivated researchers to use them for fusing multiview data [3, 112, 140]. The primary advantage of neural networks is their ability to model very complex nonlinear correlations that exist between multiple views. Early insights into their use for multiview data were provided by Yuhas et al. [163] who trained a network to predict audio using visual input. Subsequently, Cutler et al. [39] proposed to learn audiovisual correlations for the task of speaker detection using a time-delayed neural network (TDNN). Recently, various multimodal autoencoder architectures for learning shared representations have been proposed, even for the case where only a single view is present at training and testing time [112]. Another interesting work extends CCA to learning two deep encodings, one for each view, such that their correlation is maximized [3]. Regularized deep neural networks [158] have also been proposed to construct shared representations taking into account the feature inter-relationships. Each of these methods has been developed independently in different settings. Their application to event analysis and detection still remains to be explored. This is a rapidly growing area of research; we refer the interested reader to [56, 68] for recently proposed multimodal fusion architectures.

³Matlab implementations are available online at <http://plato.telecom-paristech.fr/publi/26108/>.

9.3.2 Decision-Level Integration/Fusion

Decision-level fusion, also known as *late integration* refers to the idea of combining intermediate decisions, i.e., partial classifier-outputs, in order to achieve a more accurate multimodal characterization of a content, an idea which has been explored extensively, under various configurations. This can be seen as a particular case of *ensemble learning* [125] where the base classifiers (to be combined) operate on different views of the data.

Numerous works rely on *majority voting* procedures whereby final global decisions are made based on a weighted sum of individual voters, each typically corresponding to a decision taken on a particular view. The weights are often chosen using either heuristics or trial-and-error procedures (see, for example, [93]). This idea can be better formalized using a Bayesian framework that allows for taking into account the uncertainty about each classifier decisions [71, 103].

9.3.2.1 Probabilistic Combination Rules

When using classifiers providing local probabilistic outputs $p(G_c | \mathbf{o}_{m,t})$ for the t -th observation of the m -th view, $\mathbf{o}_{m,t}$, a simplistic decision strategy assumes feature-vector observations from different views to be independent, and the decision rule consequently takes the form:

$$\hat{G} = \arg \max_c \log[p(G_c | \mathbf{o}_{0,t}, \dots, \mathbf{o}_{M-1,t})] = \arg \max_c \sum_{m=0}^{M-1} \log p(G_c | \mathbf{o}_{m,t}). \quad (9.9)$$

It is worth mentioning that alternative simple combination rules have also been employed that are discussed in Kittler et al. [86].

The previous approach does not allow for incorporating prior knowledge about the dependency structure in the data, in particular the cross-modal and temporal dependencies. To this end, sophisticated dynamic classifiers have been utilized, ranging from variants of (multistream) hidden Markov Models (HMM) [7, 60, 85, 111], through more general dynamic Bayesian networks [32, 59, 109], to even more general graphical models such as conditional random fields (CRF) [20].

9.3.2.2 Neural Networks

Neural networks can also be used for late integration. Some works have utilized them to adaptively learn the weights for fusing multiple classifiers or system outputs [74, 113]. This is typically carried out by training the network to minimize the error between estimated and oracle weights [74]. Besides, in order to take into account the temporal and multiview dependencies, a frequently used strategy is to perform end-to-end training with “fusion” integrated as a layer (usually close to the output

layer) into the architecture [49, 82]. Such methods cannot be termed as late fusion in the conventional sense as feature learning and decision fusion stages are not independent.

9.3.2.3 Other Methods

Another widespread strategy consists in using the monomodal classifier-outputs as features, on the basis of which a new classifier, that is expected to optimally perform the desired multimodal fusion, is learned [13, 156]. Also, solutions to deal with the potential imprecision of some views have been proposed using the *Dempster–Shafer* theory [54].

Finally, it is important to note that the techniques described in this section are not mutually exclusive: in practice one may jointly consider different integration strategies for different features and views (possibly being driven by some expert knowledge), and different analysis time-horizons. This raises the difficult issue of effectively and efficiently exploiting, at the final prediction stage, heterogeneous representations: low-level instantaneous features, possibly over varying time-scales, intermediate prediction results—sometimes seen as outputs of *event* or *concept* detectors—bags-of-words or bags-of-systems extracted over longer texture-windows, *etc.*

9.4 Audiovisual Event Detection

9.4.1 Motivation

The target of audiovisual event detection (AVED) is to detect specific events that occur in an audiovisual recording or real-time stream, and to identify the class of those events. Though the task is more widely addressed through the analysis of the video images, information conveyed by the sound track may become key for a proper detection. Indeed, the visual information may not be sufficient since occlusions may occur and events may be localized in space, hence not visible in the images, given that the camera field of view is necessarily restricted. Also the images may not be usable because of poor lighting conditions, or fast camera motion. AVED then enables a more reliable detection of these events, by combining audio and visual cues.

9.4.1.1 Examples in Video Content Analysis and Indexing

Researchers continue to explore various techniques for improving video content analysis and indexing for better navigation and user experience. In this context,

AV event analysis at various levels of granularity provides useful insights into the composition of such data in terms of objects, activities, and scenes. This not only improves retrieval but also provides a representation closer to our understanding of the physical world. For example, a user could search a database for activity videos such as “dribbling a basketball” or “playing a violin.” Evidently, these are two very distinct tasks where the differences can be readily detected based on auditory and motion information. Moreover, joint analysis could reveal the presence of various objects (e.g., violin, basketball) and also the surroundings (e.g., concert hall, court).

Such an analysis makes object detection and segmentation [72], concept classification [75, 76, 78], scene segmentation and change detection [149], activity analysis, and various other related tasks possible. Several systems submitted to TRECVID⁴ video content analysis tasks of multimedia event detection, story segmentation, and search rely on AV analysis [2, 77, 156].

9.4.1.2 Examples in AV Surveillance and Robot Perception

Video has recently become an increasingly important resource for forensics and surveillance [104, 124]. Video captured by CCTV systems or video recorded from mobile devices (and possibly shared on multimedia platforms) can provide essential clues in solving criminal cases. For example, when considering an investigation about a missing person, video documents can help to localize the missing person or a suspect, providing crucial information about their whereabouts. The analysis of videos linked with a missing person or her/his social network can also help to understand the conditions of the disappearance (was it a kidnapping, a runaway, etc.) and largely influence the investigation.

An investigator looking for a video in a large dataset may want to retrieve information based on the type of scene where the video was recorded or also, at a finer granularity level, based on specific events that occurred during the recording. In addition, the detection of specific events can help to confirm (or deny) the fact that a video was recorded in a particular scene. Some events are indeed representative of particular scenes. For example, train noise in all probability indicates the scene takes place in a train station. Plates and cutlery noises indicate the scene is probably taking place in a restaurant [22]. On the other hand, some events are unlikely to happen in particular scenes. AVED can then help tracking anomalies to detect abnormal events (gunshots, crowd panic, etc.) [97] or to identify a recording scene where information has voluntarily been concealed. This is the case, for example, when a kidnapper sends a ransom video recorded from inside a building but a church bell or a train passing nearby can be heard during the video. This type of information that is not present visually can help to localize the place where the video was recorded [136].

⁴TREC Video Retrieval Evaluation: <http://www-nlpir.nist.gov/projects/trecvid/>.

9.4.2 AV Event Detection Approaches

9.4.2.1 AV Event Detection and Concept Classification

Approaches to AV event detection have been very varied and data dependent. Many works for traditional event detection utilize Markov model variants such as the duration dependent input–output Markov model (DDIOMM)[110], multistream HMM, or coupled HMM [69]. The former uses a decision-level fusion strategy and the latter two do it at an intermediate level. These methods have been shown to perform better than single modality-based approaches with coupled-HMMs being particularly useful for modeling AV asynchrony.

Specifically, with regard to event detection in surveillance videos, Cristiani et al. [37] propose to use the AV concurrence matrix to identify salient events. The idea is to model the audio/video foreground and construct this matrix based on the assumption that simultaneously occurring AV foreground patterns are likely to be correlated. Joint AV analysis has also been employed extensively for sports video analysis and for broadcast analysis in general. In one approach, several feature detectors are built to encode various characteristics of field sports. Their decisions are then combined using a support vector machine (SVM)[127]. Several approaches for structuring TV news videos have also been proposed.

On the other hand, *joint codebook-based approaches* have been quite popular for the task of multimedia concept classification.⁵ In essence, each element of these multimodal codebooks captures some part of a salient AV event. Work on short-term audiovisual atoms (S-AVA) [78] aims to construct a codebook from multimodal atoms which are a concatenation of features extracted from tracked short-term visual-regions and audio. To tackle the problem of video concept classification, this codebook is built through multiple instance learning. Following this work, AV *grouplets* (AVG) [76] were proposed, where separate dictionaries are constructed from coarse audio and visual foreground/background separation. Subsequently, AVGs are formed based on the mixed-and-matched temporal correlations. For instance, an AVG could consist of frames where a basketball player is seen in the foreground with the audio of the crowd cheering in the background. As an alternative, Jhuo et al. [75] determine the relations between audio and visual modalities by constructing a bi-partite graph from their bag-of-words representation. Subsequently, spectral clustering is performed to partition and obtain *bi-modal* words. Unlike S-AVA and bimodal words, AVG has the advantage of explicitly tackling temporal interactions. However, like S-AVA, it relies on video region tracking, which is quite difficult for unconstrained videos.

⁵Here the term “concept classification” refers to generic categorization in terms of scene, event, object, or location [78].

9.4.2.2 AV Object Localization and Extraction

AV object localization and extraction refers to the problem of identifying sources visually and/or aurally. This section serves to show how objects responsible for audiovisual events can be extracted from either of the modalities through joint analysis. The general approach is to first associate the two modalities using methods discussed in Sect. 9.3. The parameters learned during the former step can then be utilized for object localization and segmentation (visual part), audio source separation (audio counterpart), or unsupervised AV object extraction in both modalities. We now discuss approaches to each of these application scenarios.

Object localization and segmentation has been a popular research problem in the computer vision community. Various approaches have leveraged the audio modality to better perform this task with the central idea of associating visual motion and audio. Fisher et al. [51] proposed to use joint statistical modeling to perform this task using mutual information. Izadinia et al. [72] consider the problem of moving-sounding object segmentation, using CCA to correlate audio and visual features. The video features consisting of mean velocity and acceleration computed over spatio-temporal segments are correlated with audio. The magnitude of the learned video projection vector indicates the strength of association between corresponding video segments and the audio. Several other works have followed the same line of reasoning while using different video features to represent motion [84, 138]. Effectiveness of CCA can be illustrated with a simple example of a video with a person dribbling a basketball [72] (see Fig. 9.3). Simplifying Izadinia et al.'s [72] visual feature extraction methodology, we compute the optical flow and use mean velocity calculated over 40×40 blocks as the visual representation and mel-spectra as the audio representation. The heat map in Fig. 9.3 shows correlation between each image block and audio. Areas with high correlation correspond to regions with motion. If we instead use a soft co-factorization model [134], it is indeed possible to track the image blocks correlated with the audio in each frame.

Another approach worth mentioning is one that uses Gestalt principles for locating sound sources in videos [105]. Inspired by Gestalt principle of temporal

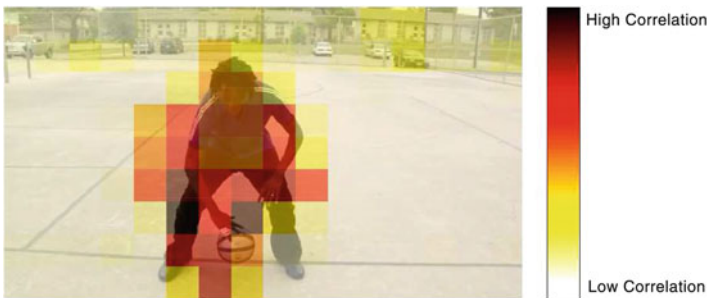


Fig. 9.3 CCA illustration: heat map showing correlation between video image regions and audio. *Black squares indicate highest correlation*

proximity the authors propose to detect synchronous audiovisual events. A particularly different approach was taken by Casanovas et al. [29] who proposed an audiovisual diffusion coefficient to remove information from video image parts which are not correlated with the audio.

Audio source separation is the audio counterpart of the previously discussed problem. The aim is to extract sound produced by each source using video information. As done for videos, mutual information maximization has been used to perform source separation in a user-assisted fashion by identifying the source spatially. Recent methods perform this within the NMF-based source separation framework [120, 132].

Several other approaches deal with both object segmentation and source separation *together* in a completely unsupervised manner. Work by Barzeley et al. [11] considers onset coincidence to identify AV objects and subsequently perform source separation. A particular limitation of this method is the requirement of setting multiple parameters for optimal performance on each example. Blind AV source separation work has also been attempted using nonnegative CCA [138] and sparse representations [28]. Independent component analysis over concatenated features from both modalities also extracts meaningful audiovisual objects [139]. However its application is limited to static scenes. Finally, multimodal dictionary learning has also been utilized in this context [94].

While the methods discussed in this section have been shown to work well in controlled environments, their performance is expected to degrade in dense audiovisual scenarios. Moreover, they make a simplifying assumption that all the objects are seen onscreen. It must be emphasized that most of these techniques can be considered symmetric, in the sense that they can be applied to tasks in either of the modalities with appropriate representations.

9.5 Microphone Array-Based Sound Scene Analysis

In complex sound scenes the sounds coming from different sources can be overlapping in time and frequency. Single channel processing can discriminate sources based on time or frequency as long as they are separated in either time or frequency. Trying to detect or classify sound events that are overlapping both in time and frequency directly from a single channel signal will generally result in a confusion between events. An alternative approach is to attempt to separate individual events prior to detection or classification. However, trying to separate sounds that are overlapping both in time and frequency with single channel techniques is known to be problematic and will inevitably introduce a loss of information resulting in a degradation of the subsequent detection and classification performance. *Microphone arrays* enable the usage of multichannel techniques that exploit not only temporal and spectral diversity between sources but also spatial information about their location.

Historically microphone arrays were composed of a set of microphones placed along a straight line (with constant spacing between microphones for linear arrays or variable spacing for logarithmic arrays). Less constraint arrays have been used for specific purposes such as spherical and circular arrays and more recently arrays without spatial constraints in the case of wireless acoustic sensor networks became a popular research topic. Some approaches and concepts presented here are applicable only to specific array topologies or at least when the topology is known beforehand (see also below). In this chapter we also assume that the signals coming from different microphones are synchronized at the stage of sampling in order to allow for the exploitation of spatial cues. Readers should keep in mind that at the time of writing of this book, dealing with unsynchronized microphone arrays is still an open research problem.

9.5.1 Spatial Cues Modeling

In order to exploit spatial information about the sound sources, audio scene analysis algorithms usually first model the spatial cues and then estimate the corresponding parameters. Both deterministic and probabilistic modeling of such spatial cues have been widely considered in the literature. The former case usually relies on (a) the *point source* assumption, where sound from a source is assumed to come from a single position, and (b) the *narrowband approximation*, where a mixing process from an audio source to the microphone array is characterized by a mixing frequency dependent vector [98]. Probabilistic modeling is usually applied for reverberated or diffuse sources, where sound from a source may come from many directions due to the reverberation, e.g., source localization [18, 63, 116], separation [46, 73, 99], and beamforming systems [17, 48]. This section will discuss some typical spatial cue models, in both a deterministic and a probabilistic sense, for different audio scene analysis applications.

9.5.1.1 Binaural Approach

Humans generally combine cues from several audiovisual streams to localize sound sources spatially. The main cues for localization in the horizontal hemisphere are related to binaural hearing (relying on the difference between the signal reaching the right ear and the signal reaching the left ear). All these cues are encoded in the so-called interaural transfer function (ITF) that includes the following:

- The **interaural time difference (ITD)** is the difference between the time-of-arrival of a signal at the left ear and the right ear. It is useful to localize sounds based on their onset and at low frequency (below 1.5 kHz) [89] (see also Fig. 9.4a).

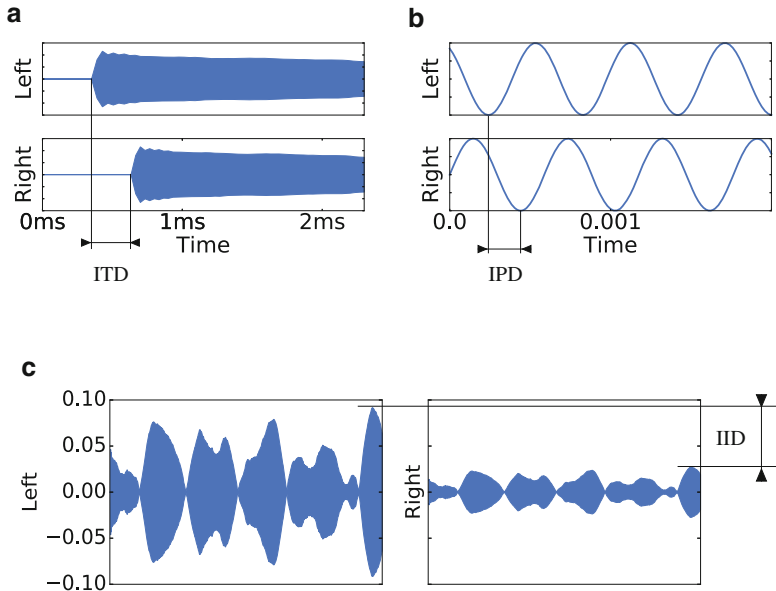


Fig. 9.4 Artificial representation of the binaural cues. (a) ITD, (b) IPD, (c) IID

- The **interaural phase difference (IPD)** is the phase difference between the signal at the left ear and the right ear. It is useful to localize on-going sound as long as the wavelength is larger than the diameter of the head (below 1.5 kHz) [162] (see also Fig. 9.4b);
- The **interaural intensity difference (IID)** is the difference in level between the signal at the left ear and the right ear due to the acoustic shadow produced by the head for sounds above 3 kHz (below the so-called head shadow effect is not present) [108] (see also Fig. 9.4c).

All the concepts mentioned above can be extended to general microphone array setups. The ITD and IPD concepts directly generalize to linear microphone arrays where they relate rather straightforwardly to time difference of arrival (TDOA) and direction of arrival (DOA). In this case, however, the arrays have to be designed carefully to prevent spatial aliasing. The IID concept is less applicable to small linear arrays as it relies on the head shadow effect. Indeed, in small arrays the level difference between the signal impinging two consecutive microphones might not be significant. However, in ad-hoc arrays where the topology is unconstrained, the microphones can be quite far apart and IID can become insightful as well, granted that the microphone positions are known beforehand. These spatial cues are extensively exploited to extract a signal of interest from the mixture using beamforming approaches described in Sect. 9.5.1.2 (for example, the delay-and-sum beamformer directly relies on ITD). Spatial cues can also be used directly for sound source localization (see also Sect. 9.5.2.3) and, by proxy, for source separation (see also Sect. 9.5.2.1) and sound event detection (see also Sect. 9.5.2.2).

9.5.1.2 Beamforming Methods

Fixed beamformers compose a first simple class of multichannel algorithms which can separate signals coming from different directions. A fixed beamformer tries to steer toward the direction from where the desired sound signal comes and to reject signals coming from other directions. The main categories of fixed beamformers include delay-and-sum beamformers, filter-and-sum beamformers [66], superdirective microphone arrays [36], or the original formulation of the minimum variance distortionless beamformer (MVDR) [26].

Adaptive beamformers try to steer toward the direction of the desired sound signal and to adaptively minimize the contributions from the undesired sources coming from other directions. This typically yields a constrained optimization problem. Frost introduced the linearly constrained minimum variance beamformer (LCMV) as an adaptive framework for MVDR [55].

The generalized side lobe canceler (GSC), also known as the Griffiths-Jim beamformer, is an alternative approach to the LCMV where the optimization problem is reformulated as an unconstrained problem [62]. The GSC can be decomposed as a fixed beamformer steering toward the desired source, a blocking matrix, and a multichannel adaptive filter [65].

The multichannel Wiener filters (MWF) represent another class of multichannel signal extraction algorithms which are defined by an unconstrained optimization problem [45]. MWF-based algorithms can be implicitly decomposed into a spatial filter and a spectral filter, and can indeed be considered as beamformers [135]. Besides, a reformulation of MWF allows for explicitly controlling the spectral distortion introduced [45, 135].

9.5.1.3 Nonstationary Gaussian Model

The nonstationary Gaussian framework has emerged in audio source separation [46, 50, 114, 119] as a probabilistic modeling of the reverberated sources. It was then also applied in, e.g., multichannel acoustic echo cancellation [144] and multichannel speech enhancement [145]. In this paradigm, the short-time Fourier transform (STFT) coefficients of the *source images* $\mathbf{c}_j(t, f)$, i.e., the contribution of the j -th source ($1 \leq j \leq J$) at the microphone array, are modeled as a zero-mean Gaussian random vector whose covariance matrix $\widehat{\mathbf{R}}_j(t, f) = \mathbb{E}(\mathbf{c}_j(t, f)\mathbf{c}_j^H(t, f))$ can be factorized as

$$\widehat{\mathbf{R}}_j(t, f) = v_j(t, f)\mathbf{R}_j(t, f), \quad (9.10)$$

where $v_j(t, f)$ are scalar time-varying *variances* encoding the spectro-temporal power of the sources and $\mathbf{R}_j(t, f)$ are $I \times I$ *spatial covariance matrices* encoding their spatial position and spatial width. This model does not rely on the point source assumption nor on the narrowband assumption, hence it appears applicable to

reverberated or diffuse sources. In the general situation where the sound source can be moving, the spatial cues encoded by $\mathbf{R}_j(t, f)$ are time-varying. However, in most cases where the source position is fixed and the reverberation is moderate, the spatial covariance matrices are time-invariant: $\mathbf{R}_j(t, f) = \mathbf{R}_j(f)$. Different possibilities of parameterizing $\mathbf{R}_j(f)$ have been considered in the literature resulting in either the rank-1 or the full-rank matrices, where the later case was shown to be more appropriate for modeling the reverberated and diffuse sources as it accounts directly for the interchannel correlation in the off-diagonal entries of $\mathbf{R}_j(f)$ [46].

9.5.2 Spatial Cues-Based Sound Scene Analysis

This section will discuss the use of spatial cue models presented in the previous section in some specific applications, namely sound source separation, acoustic event detection, and moving sound source localization and tracking.

9.5.2.1 Sound Source Separation

In daily life, recorded sound scenes often result from the superposition of multiple sound sources which prevent both human and machines from well localizing and perceiving the target sound sources. Thus, source separation plays a key role in sound scene analysis, and its goal is to extract the signals of individual sound sources from an observed mixture [98]. It offers many practical applications in, e.g., communication, hearing aids, robotics, and music information retrieval [6, 14, 100, 152].

Most source separation algorithms operate in the time-frequency (T-F) domain with the mixing process formulated as

$$\mathbf{x}(t, f) = \sum_{j=1}^J \mathbf{c}_j(t, f) \quad (9.11)$$

where $\mathbf{x}(t, f) \in \mathbb{C}^{I \times 1}$ denotes the STFT coefficients of the I -channel mixture at T-F point (t, f) , and $\mathbf{c}_j(t, f) \in \mathbb{C}^{I \times 1}$ is the j -th source image. As $\mathbf{c}_j(t, f)$ encodes both spectral information about the sound source itself and the spatial information about the source position, a range of spectral and spatial models has been considered in the literature resulting in various source separation approaches. In the determined case where $I \geq J$, non-Gaussian modeling such as frequency-domain independent component analysis (FDICA) has been well-studied [122, 128]. In the under-determined situation where $I < J$, sparse component analysis (SCA) has been largely investigated [19, 61, 81]. As a specific example of the nonstationary Gaussian modeling presented in Sect. 9.5.1.3, the parameters are usually estimated by the expectation maximization (EM) algorithm derived in either the maximum

likelihood (ML) sense [46] or the maximum a posteriori (MAP) sense [47, 117, 119]. Then source separation is achieved by the multichannel Wiener filtering. Readers are referred to, e.g., [95, 150] for the survey of recent advances on both *blind* scenarios and *informed* scenarios which exploit some prior knowledge about the sources themselves [119] or the mixing process [47] to better guide the source separation.

9.5.2.2 Sound Event Detection

As different sound events usually occur at different spatial locations in the sound scene, spatial cues obtained from microphone array processing intrinsically offer important information for SED. As an example, information about the source directions inferred from the interchannel time differences of arrival (TDOA) was used to help partitioning home environments into several areas containing different types of sound events in [151]. The combination of these spatial features with the classic MFCC was reported to improve the event classification in the experiment. Motivated by binaural processing, in [1] the stereo log-mel-band energy is extracted from stereo recordings to train the neural networks in order to obtain a meaningful cue similarly to the IID.

9.5.2.3 Localization and Tracking of Sound Sources

Sound source localization and tracking are concerned with estimating and following the position of a target source within a sound scene. This active field of research in microphone array processing finds important applications, e.g., in surveillance or video conferencing where the camera should be able to follow the moving speaker, and even can automatically switch the capture to an active sound source in multiple source environments [153]. Spatial cues offered by the multichannel audio capture play a key role in deriving the algorithms.

The problem of acoustic source localization has been a relevant topic in the audio processing literature for the past three decades because of its applicability to a wide range of applications [41, 146]. The most effective solutions rely on the use of spatial distributions of microphones, which sample the sound field at several locations. Spurious events, reverberation, and environmental noise, however, can be a significant cause of localization error. In order to ease the problem, at least for those errors that are contained in a limited number of time frames, source tracking techniques can come in handy, as they are able to perform trajectory regularization, even on the fly. Typical approaches are based on particle [5, 91, 155], Kalman [4, 57], or distributed Kalman filtering [143].

Different methodologies have been developed for the localization of acoustic sources through microphone arrays. Those that gained in popularity are based on measurements of the time delay between array microphones. Working on the time domain is often a suitable choice for wideband signals, and most techniques tend to

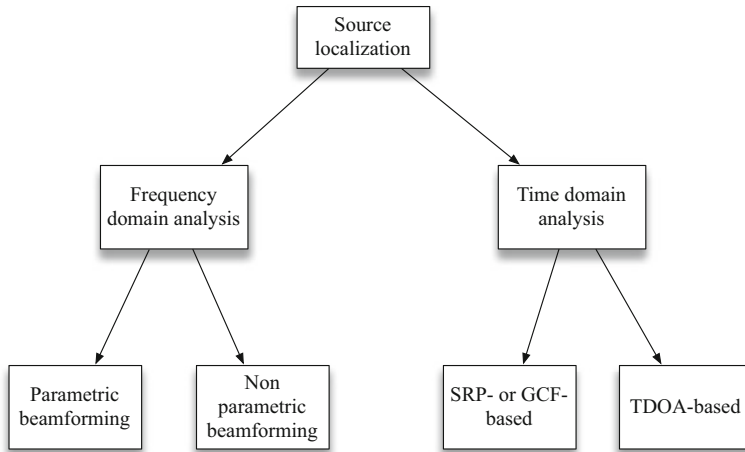


Fig. 9.5 Taxonomy of source localization techniques

rely on the analysis of the generalized cross-correlation (GCC) of the signals [27] and variants thereof. Localization in the frequency domain, however, can be shown to attain good results for narrowband or harmonic sources immersed in a wideband noise and rely on the analysis of the covariance matrix of the array data. A taxonomy of the localization techniques is represented in Fig. 9.5.

Time-Domain Localization

Steered response power (SRP, [42, 43, 102]) and global coherence field (GCF, [115]) proceed through the computation of a coherence function that maps the GCC values at different microphone pairs on the hypothesized source location. A source location estimate is found as the point in space that maximizes the coherence function. In [23] the scenario of multiple sources is accommodated through a two-step procedure that, after localizing the most prominent source, deemphasizes its contribution in the GCC, so that other sources can be localized. These techniques are known for their high level of accuracy, and are suitable for networks of microphone arrays, where synchronization can only be guaranteed between microphones of the same array. One limitation of such solutions is their computational cost, which is proportional to the number of hypothesized source locations. This means that increasing the spatial resolution results in higher computational costs. Some solutions have been proposed in the literature to mitigate this problem. In [165] the authors propose a hierarchical method that begins with a coarser grid, and refines the estimate at different steps by computing the map for finer grids concentrated around the candidate locations estimated at the previous step. In [44] a similar approach is adopted, but a stochastic region contraction strategy is used for going from a coarser to a finer grid. An example of steered response power with stochastic region contraction map is shown in Fig. 9.6.

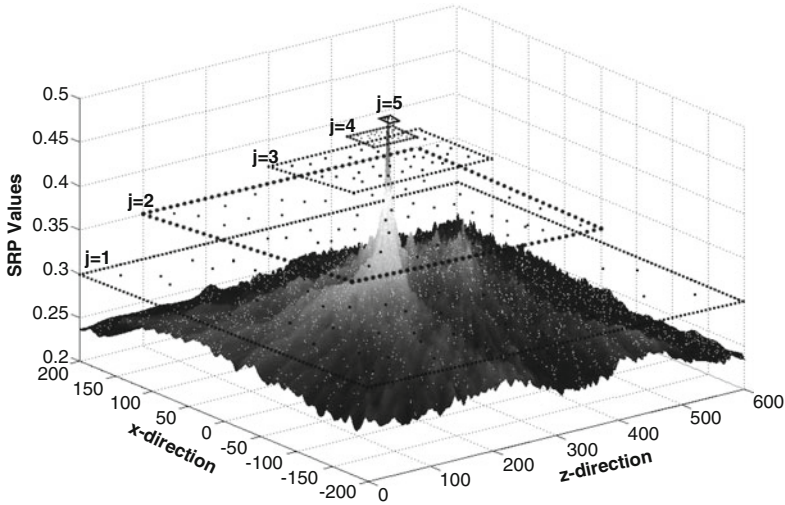


Fig. 9.6 Example of a coherent map using the Steered Response Power with Stochastic Region Contraction technique (SRP-SRC, from [44])

Less cumbersome are the solutions based on the time difference of arrival (TDOA), which is estimated as the time lag of the GCC that exhibits the maximum value. The TDOA is then converted into range difference (RD), which measures the difference of the range between the source and the two microphones in the pair. The locus of candidate source locations corresponding to a given TDOA is a branch of hyperbola whose foci are in the microphones locations, and whose aperture is proportional to the measured TDOA. The most straightforward technique for localization consists in intersecting branches of hyperbolas corresponding to the TDOA measurements coming from different pairs of microphones. The cost function that is based on this procedure is strongly nonlinear, which makes the method sensitive to measurement errors. Least squares cost functions provide a good approximation [12, 34, 70, 129]. The main drawback of TDOA-based localization is its sensitivity to outlier measurements. In [24, 25, 130] techniques for removal of the outliers were presented. In particular, the DATEMM algorithm [130] is based on the observation that TDOAs over a closed loop must sum to zero.

Frequency-Domain Localization

Techniques in the frequency domain are based on the observation that different microphones in the array will receive differently delayed replicas of the source signals. This, in the frequency domain, corresponds to a phase offset. For distant sources the phase offset between adjacent microphones is constant throughout the array. Delay-and-sum beamformers compensate the offsets so that the components

related to a direction will sum up coherently and the others will not. The estimation of the direction of arrival (DOA) of the target source proceeds by searching for the direction that maximizes the output energy of the beamformer over a grid of directions [141, Chapter 6]. The most straightforward nonparametric beamformer is the delay and sum, which is known for its low resolution capabilities, making it difficult to distinguish sources that are seen under close angles from the array viewpoint. The minimum variance distortionless response beamformer (MVDR, [26]) partially improves the resolution capabilities. Parametric techniques, among which it is worth mentioning multiple signal classification (MUSIC, [131]), and estimation of the signal parameters through rotational invariance techniques (ESPRIT, [126]) bring improvements in terms of resolution. However, they are known for their sensitivity to noise and reverberation, which tends to introduce spurious localizations. The superdirective data-independent beamformer [16] was shown to partially mitigate this problem. An interesting solution to the sensitivity to reverberation was proposed in [137] for the detection of gunshots using networks of sensors, each equipped with four or more microphones. For each sensor, both DOA and TDOA are measured. Source location is estimated by intersecting the loci of potential source locations (hyperbolas and direction of arrival) for the two kind of measurements from all the sensors. In reverberant conditions and in the presence of interferers, some TDOAs and some DOAs could be related to spurious paths, thus providing multiple estimates of the gunshot location. The actual gunshot location is found as the one that maximizes the number of consistent TDOAs and DOAs.

It is important to notice that TDOA-based and frequency-domain source localization techniques require the synchronization of the microphones within the array. This, in fact, becomes an issue when multiple independent small arrays are deployed in different locations. In [25] the authors propose a technique for the localization without requiring a preliminary synchronization of the arrays by including the time offsets between the arrays into the unknowns, along with the location of the source. Another important issue is the self-calibration of the array, i.e., the estimation of the mutual relative positions of the microphones [38, 147]. The widespread diffusion of mobile phones and devices equipped with one or more microphones enables the implementation of a wireless acoustic sensor network in seconds, for goals ranging from teleconferencing to security. In this context, however, both calibration and synchronization are needed before normal operation [123].

Acoustic Source Tracking

Independently of the adopted localization method, reverberation and interferers could introduce spurious localizations. The goal of source tracking is to alleviate the influence of outliers. The idea behind tracking is that measurements related to the actual source must follow a dynamical model whereas those related to spurious sources must not [155]. Another goal that can be pursued with tracking systems is that of fusing information coming from both audio and visual localization systems [9, 142]. Several solutions have been presented in the literature. The Kalman filter

[57] is a linear system characterized by two equations. The state equation models the evolution of the state of a system (location and speed of the source) from one time frame to the next one. The observation equation links the state variables with the observable measurements. The goal of the Kalman filter is to estimate the current state from the knowledge of time series of the observations.

Recently, distributed Kalman filters have been used, which enable the tracking of acoustic sources also in the case of distributed array networks [164], without requiring that all nodes communicate the whole state of the system.

Inherent assumptions that lie in the use of the Kalman filter are the linearity and Gaussianity of measurement and state vectors. In order to gain in robustness against the nonlinearity, the use of the extended Kalman filter has been proposed [142], which linearizes the nonlinear system around the working point. In order to gain in robustness against non-Gaussian conditions, however, one has to resort to a different modeling of the source dynamics. In recent years particle filter gained interest in the source localization community due to the fact that it is suitable also to perform tracking in nonlinear non-Gaussian systems and, more in general, for its higher performance [155]. Particle filtering [8] assumes that both state and measurement vectors are known in a probabilistic form. Once a new measurement vector is available, the likelihood function of the current observation from a given state is sampled through particles. Each particle is assigned a weight, which determines its relevance in the likelihood function. Only relevant particles will be propagated to the next step. The source location is determined as the centroid of the set of particles. An example of tracking of one, two, or three acoustic sources on a given trajectory for DOA measurements is shown in Fig. 9.7.

In audio surveillance contexts, it is important to enable localization also when multiple sources are active at any time, with a small convergence time when acoustic sources alternate. This is important, for example, in events that involve multiple acoustic sources (brawls, people yelling, etc.). In recent years, swarm particle filtering has shown to address this scenario particularly well [121]. It is based on the idea that the propagation of each particle to the next step is determined not only by the previous history of the particle itself, but also by the particle that exhibits the best likelihood at the current time instant. Consequently, the overall behavior of the systems resembles that of a bird flock, rapidly moving toward the active source. An example of behavior of swarm particle filtering is shown in Fig. 9.8. Here two sets of particles at four consecutive time frames estimate the location of a source using particle filtering (PF) and swarm particle filtering (Swarm). The two sets are initialized identically. It is possible to notice that after four steps, the swarm particles cluster around the source location, while the PF is still converging.

9.6 Conclusion and Outlook

Multichannel and multimodal data settings represent opportunities to address complex real-world scene and event classification problems in a more effective manner. The availability of concurrent, hence potentially complementary streams of data is amenable to a more robust analysis, by effectively combining them, using

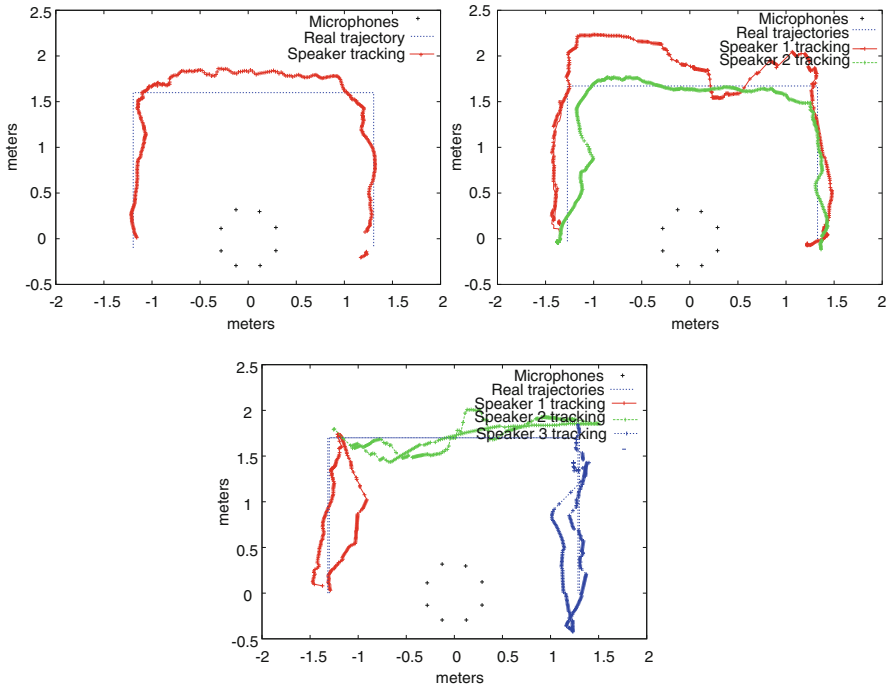
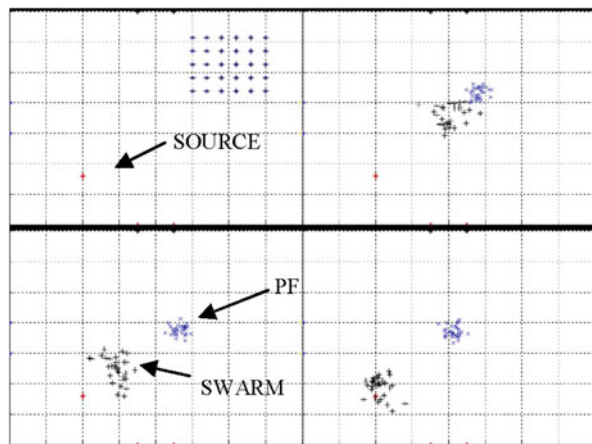


Fig. 9.7 Example of tracking of one, two, or three sources over a prescribed trajectory (from [148])

Fig. 9.8 Example of behavior of two sets of particles propagated using particle filtering (PF) and swarm particle filtering (SWARM). The two sets of particles occupy the same location at the first time frame (from [121])



appropriate techniques, be it at the input representation-level, the feature-level, or the decision-level. Successful applications of such techniques have been realized in various multichannel audio and audiovisual scene analysis tasks.

Yet, a number of research questions remain open in these settings. Notably, it is still not clear how to generically detect when some of the data views are temporarily not reliable (typically noisy or out of focus, with respect to the classes of interest) and which strategies should be developed that can efficiently ignore such views and proceed with the classification (or any other similar data processing) using models which were perhaps trained assuming all views are available.

Also, given the complexity of accurately annotating all data views, especially for instantaneous multi-label event classification tasks, that is when multiple events may occur simultaneously, it is important to consider learning methods that can take advantage of very coarse ground-truth labels, which may have been obtained based on just one of the views, without necessarily being relevant for others. An example of this is the “blind” annotation of the audio track of a video (without considering the images) where sound events may not be visible onscreen at the same time stamps. Multiple instance learning and weakly supervised learning techniques may turn out to be effective learning paradigms to address these difficulties.

References

1. Adavanne, S., Parascandolo, G., Pertila, P., Heittola, T., Virtanen, T.: Sound event detection in multichannel audio using spatial and harmonic features. In: Proceedings of the IEEE AASP Chall Detect Classif Acoust Scenes Events (2016)
2. Amir, A., Berg, M., Chang, S.F., Hsu, W., Iyengar, G., Lin, C.Y., Naphade, M., Natsev, A., Neti, C., Nock, H., et al.: Ibm research trecvid-2003 video retrieval system. In: NIST TRECVID-2003 (2003)
3. Andrew, G., Arora, R., Bilmes, J.A., Livescu, K.: Deep canonical correlation analysis. In: Proceedings of the International Conference on Machine Learning (2013)
4. Antonacci, F., Lonoce, D., Motta, M., Sarti, A., Tubaro, S.: Efficient source localization and tracking in reverberant environments using microphone arrays. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 4, pp. iv–1061. IEEE, New York (2005)
5. Antonacci, F., Matteucci, M., Migliore, D., Riva, D., Sarti, A., Tagliasacchi, M., Tubaro, S.: Tracking multiple acoustic sources in reverberant environments using regularized particle filter. In: Proceedings of the International Conference on Digital Signal Processing, pp. 99–102 (2007)
6. Arai, T., Hodoshima, H., Yasu, K.: Using steady-state suppression to improve speech intelligibility in reverberant environments for elderly listeners. *IEEE Trans. Audio Speech Lang. Process.* **18**(7), 1775–1780 (2010)
7. Argones Rúa, E., Bredin, H.H., García Mateo, C., Chollet, G.G., González Jiménez, D.: Audio-visual speech asynchrony detection using co-inertia analysis and coupled hidden Markov models. *Pattern Anal. Appl.* **12**(3), 271–284 (2008)
8. Arulampalam, M., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **50**(2), 174–188 (2002)
9. Asoh, H., Asano, F., Yoshimura, T., Yamamoto, K., Motomura, Y., Ichimura, N., Hara, I., Ogata, J.: An application of a particle filter to Bayesian multiple sound source tracking with audio and video information fusion. In: Proceedings of the Fusion, pp. 805–812. Citeseer (2004)

10. Atrey, P.K., Hossain, M.A., El Saddik, A., Kankanhalli, M.S.: Multimodal fusion for multimedia analysis: a survey. *Multimed Syst* **16**(6), 345–379 (2010)
11. Barzelay, Z., Schechner, Y.Y.: Harmony in motion. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8 (2007)
12. Beck, A., Stoica, P., Li, J.: Exact and approximate solutions of source localization problems. *IEEE Trans. Signal Process.* **56**(5), 1770–1778 (2008)
13. Benmokhtar, R., Huet, B.: Neural network combining classifier based on Dempster-Shafer theory for semantic indexing in video content. In: *International MultiMedia Modeling Conference (MMM 2007)*, Singapore, 9–12 January 2007. LNCS, vol. 4352/2006, Part II. <http://www.eurecom.fr/publication/2119>
14. Bertin, N., Badeau, R., Vincent, E.: Enforcing harmonicity and smoothness in Bayesian non-negative matrix factorization applied to polyphonic music transcription. *IEEE Trans. Audio Speech Lang. Process.* **18**(3), 538–549 (2010)
15. Bießmann, F., Meinecke, F.C., Gretton, A., Rauch, A., Rainer, G., Logothetis, N.K., Müller, K.R.: Temporal kernel cca and its application in multimodal neuronal data analysis. *Mach. Learn.* **79**(1–2), 5–27 (2010)
16. Bitzer, J., Simmer, K.U.: Superdirective microphone arrays. In: *Microphone Arrays*, pp. 19–38. Springer, New York (2001)
17. Bitzer, J., Simmer, K.U., Kammeyer, K.D.: Theoretical noise reduction limits of the generalized sidelobe canceller (GSC) for speech enhancement. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2965–2968 (1999)
18. Blandin, C., Ozerov, A., Vincent, E.: Multi-source TDOA estimation in reverberant audio using angular spectra and clustering. *Signal Process.* **92**(8), 1950–1960 (2012)
19. Bofill, P., Zibulevsky, M.: Underdetermined blind source separation using sparse representations. *Signal Process.* **81**(11), 2353–2362 (2001)
20. Bousmalis, K., Morency, L.P.: Modeling hidden dynamics of multimodal cues for spontaneous agreement and disagreement recognition. In: *International Conference on Automatic Face & Gesture Recognition*, pp. 746–752 (2011)
21. Bredin, H., Chollet, G.: Measuring audio and visual speech synchrony: methods and applications. *Proceedings of the IET International Conference on Visual Information Engineering*, pp. 255–260 (2006)
22. Bregman, A.S.: *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, Cambridge (1994)
23. Brutti, A., Omologo, M., Svaizer, P.: Localization of multiple speakers based on a two step acoustic map analysis. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4349–4352 (2008)
24. Canclini, A., Antonacci, F., Sarti, A., Tubaro, S.: Acoustic source localization with distributed asynchronous microphone networks. *IEEE Trans. Audio Speech Lang. Process.* **21**(2), 439–443 (2013)
25. Canclini, A., Bestagini, P., Antonacci, F., Compagnoni, M., Sarti, A., Tubaro, S.: A robust and low-complexity source localization algorithm for asynchronous distributed microphone networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(10), 1563–1575 (2015)
26. Capon, J.: High-resolution frequency-wavenumber spectrum analysis. *Proc. IEEE* **57**(8), 1408–1418 (1969)
27. Carter, G.C.: Coherence and time delay estimation. *Proc. IEEE* **75**(2), 236–255 (1987)
28. Casanovas, A., Monaci, G., Vanderghenst, P., Gribonval, R.: Blind audiovisual source separation based on sparse redundant representations. *IEEE Trans. Multimed.* **12**(5), 358–371 (2010)
29. Casanovas, A.L., Vanderghenst, P.: Nonlinear video diffusion based on audio-video synchrony. *IEEE Trans. Multimed.*, 2486–2489 (2010). doi:[10.1109/ICASSP.2010.5494896](https://doi.org/10.1109/ICASSP.2010.5494896)
30. Chang, S.F., Ellis, D., Jiang, W., Lee, K., Yanagawa, A., Loui, A.C., Luo, J.: Large-scale multimodal semantic concept detection for consumer video. In: *Proceedings of the International Workshop on Multimedia Information Retrieval, MIR '07*, pp. 255–264. ACM, New York, NY (2007)

31. Chibelushi, C.C., Mason, J.S.D., Deravi, N.: Integrated person identification using voice and facial features. In: Proceedings of the IEE Colloquium on Image Processing for Security Application, pp. 4/1–4/5 (1997)
32. Choudhury, T., Reh, J.M., Pavlovic, V., Pentland, A.: Boosting and structure learning in dynamic Bayesian networks for audio-visual speaker detection. In: Proceedings of the IEEE International Conference on Pattern Recognition, vol. 3, pp. 789–794 (2002)
33. Cichocki, A., Zdunek, R., Amari, S.: Nonnegative matrix and tensor factorization. *IEEE Signal Process. Mag.* **25**(1), 142–145 (2008)
34. Compagnoni, M., Bestagini, P., Antonacci, F., Sarti, A., Tubaro, S.: Localization of acoustic sources through the fitting of propagation cones using multiple independent arrays. *IEEE Trans. Audio Speech Lang. Process.* **20**(7), 1964–1975 (2012)
35. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. Wiley, London (2006)
36. Cox, H., Zeskind, R., Kooij, T.: Practical supergain. *IEEE Trans. Acoust. Speech Signal Process.* **34**(3), 393–398 (1986)
37. Cristani, M., Bicego, M., Murino, V.: Audio-visual event recognition in surveillance video sequences. *IEEE Trans. Multimed.* **9**(2), 257–267 (2007)
38. Crocco, M., Bue, A.D., Murino, V.: A bilinear approach to the position self-calibration of multiple sensors. *IEEE Trans. Signal Process.* **60**(2), 660–673 (2012)
39. Cutler, R., Davis, L.: Look who’s talking: speaker detection using video and audio correlation. In: Proceedings of the IEEE International Conference on Multimedia & Expo, vol. 3, pp. 1589–1592. IEEE, New York (2000)
40. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 886–893. IEEE, New York (2005)
41. D’Arca, E., Robertson, N., Hopgood, J.: Look who’s talking: Detecting the dominant speaker in a cluttered scenario. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2014)
42. DiBiase, J., Silverman, H., Brandstein, M.: Robust localization in reverberant rooms. In: *Microphone Arrays*, pp. 157–180. Springer, New York (2001)
43. Dmochowski, J., Benesty, J., Affes, S.: A generalized steered response power method for computationally viable source localization. *IEEE Trans. Audio Speech Lang. Process.* **15**(8), 2510–2526 (2007)
44. Do, H., Silverman, H., Yu, Y.: A real-time SRP-PHAT source location implementation using stochastic region contraction (SRC) on a large-aperture microphone array. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp. 1121–1124. IEEE, New York (2007)
45. Doclo, S., Moonen, M.: GSVD-based optimal filtering for single and multimicrophone speech enhancement. *IEEE Trans. Signal Process.* **50**(9), 2230–2244 (2002)
46. Duong, N.Q.K., Vincent, E., Gribonval, R.: Under-determined reverberant audio source separation using a full-rank spatial covariance model. *IEEE Trans. Audio Speech Lang. Process.* **18**(7), 1830–1840 (2010)
47. Duong, N.Q.K., Vincent, E., Gribonval, R.: Spatial location priors for Gaussian model based reverberant audio source separation. *EURASIP J. Adv. Signal Process.* **2013**(1), 1–11 (2013)
48. Elko, G.W.: Spatial coherence functions for differential microphones in isotropic noise fields. In: *Microphone Arrays: Signal Processing Techniques and Applications*, pp. 61–85. Springer, New York (2001)
49. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition (2016). arXiv preprint arXiv:1604.06573
50. Févotte, C., Cardoso, J.F.: Maximum likelihood approach for blind audio source separation using time-frequency Gaussian models. In: Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 78–81 (2005)
51. Fisher, J., Darrell, T., Freeman, W.T., Viola, P., Fisher III, J.W.: Learning joint statistical models for audio-visual fusion and segregation. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 772–778 (2001)

52. FitzGerald, D., Cranitch, M., Coyle, E.: Extended nonnegative tensor factorisation models for musical sound source separation. *Comput. Intell. Neurosci.* **2008**, 15 pp. (2008). Article ID 872425; doi:[10.1155/2008/872425](https://doi.org/10.1155/2008/872425)
53. Fitzgerald, D., Cranitch, M., Coyle, E.: Using tensor factorisation models to separate drums from polyphonic music. In: *Proceedings of the International Conference on Digital Audio Effects (2009)*
54. Foucher, S., Lalibert, F., Boulianne, G., Gagnon, L.: A Dempster-Shafer based fusion approach for audio-visual speech recognition with application to large vocabulary French speech. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2006)*
55. Frost, O.L.: An algorithm for linearly constrained adaptive array processing. *Proc. IEEE* **60**(8), 926–935 (1972)
56. Gandhi, A., Sharma, A., Biswas, A., Deshmukh, O.: Gethr-net: A generalized temporally hybrid recurrent neural network for multimodal information fusion (2016). arXiv preprint arXiv:1609.05281
57. Gehrig, T., Nickel, K., Ekenel, H., Klee, U., McDonough, J.: Kalman filters for audio-video source localization. In: *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 118–121. IEEE, New York (2005)
58. Goecke, R., Millar, J.B.: Statistical analysis of the relationship between audio and video speech parameters for Australian English. In: *Proceedings of the ISCA Tutor Res Workshop Audit-Vis Speech Process*, pp. 133–138 (2003)
59. Gowdy, J.N., Subramanya, A., Bartels, C., Bilmes, J.A.: DBN based multi-stream models for audio-visual speech recognition. In: *Proceedings of the IEEE International Conference of Acoustics, Speech and Signal Processing (2004)*
60. Gravier, G., Potamianos, G., Neti, C.: Asynchrony modeling for audio-visual speech recognition. In: *Proceedings of the International Conference on Human Language Technology Research*, pp. 1–6. Morgan Kaufmann Publishers Inc., San Diego (2002)
61. Gribonval, R., Zibulevsky, M.: Sparse component analysis. In: *Handbook of Blind Source Separation, Independent Component Analysis and Applications*, pp. 367–420. Academic, New York (2010)
62. Griffiths, L., Jim, C.: An alternative approach to linearly constrained adaptive beamforming. *IEEE Trans. Antennas Propag.* **30**(1), 27–34 (1982)
63. Gustafsson, T., Rao, B.D., Trivedi, M.: Source localization in reverberant environments: modeling and statistical analysis. *IEEE Trans. Speech Audio Process.* **11**, 791–803 (2003)
64. Hardoon, D.R., Szedmak, S., Shawe-Taylor, J.: Canonical correlation analysis: an overview with application to learning methods. *Neural Comput.* **16**(12), 2639–2664 (2004)
65. Haykin, S.: *Adaptive Filter Theory*, 5th edn. Pearson Education, Upper Saddle River (2014)
66. Haykin, S., Justice, J.H., Owsley, N.L., Yen, J., Kak, A.C.: *Array Signal Processing*. Prentice-Hall, Inc., Englewood Cliffs (1985)
67. Hotelling, H.: Relations between two sets of variates. *Biometrika* **28**(3–4), 321–377 (1936)
68. Hu, D., Li, X., Lu, X.: Temporal multimodal learning in audiovisual speech recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)*
69. Huang, P.S., Zhuang, X., Hasegawa-Johnson, M.: Improving acoustic event detection using generalizable visual features and multi-modality modeling. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 349–352. IEEE, New York (2011)
70. Huang, Y., Benesty, J., Elko, G., Mersereau, R.: Real-time passive source localization: a practical linear-correction least-squares approach. *IEEE Trans. Speech Audio Process.* **9**(8), 943–956 (2001)
71. Ivanov, Y., Serre, T., Bouvrie, J.: Error weighted classifier combination for multi-modal human identification. Tech. Rep. MIT-CSAIL-TR-2005-081, MIT (2005)
72. Izadinia, H., Saleemi, I., Shah, M.: Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Trans. Multimed.* **15**(2), 378–390 (2013)

73. Izumi, Y., Ono, N., Sagayama, S.: Sparseness-based 2CH BSS using the EM algorithm in reverberant environment. In: Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 147–150 (2007)
74. Jaureguiberry, X., Vincent, E., Richard, G.: Fusion methods for speech enhancement and audio source separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(7), 1266–1279 (2016)
75. Jhuo, I.H., Ye, G., Gao, S., Liu, D., Jiang, Y.G., Lee, D., Chang, S.F.: Discovering joint audio-visual codewords for video event detection. *Mach. Vis. Appl.* **25**(1), 33–47 (2014)
76. Jiang, W., Loui, A.C.: Audio-visual grouplet: temporal audio-visual interactions for general video concept classification. In: Proceedings of the ACM International Conference on Multimedia, Scottsdale, pp. 123–132. (2011)
77. Jiang, Y.G., Zeng, X., Ye, G.: Columbia-UCF TRECVID2010 multimedia event detection: combining multiple modalities, contextual concepts, and temporal matching. In: Proceedings of the NIST TRECVID-2003 (2003)
78. Jiang, W., Cotton, C., Chang, S.F., Ellis, D., Loui, A.: Short-term audiovisual atoms for generic video concept classification. In: Proceedings of the ACM International Conference on Multimedia, pp. 5–14. ACM, New York (2009)
79. Jiang, Y.G., Bhattacharya, S., Chang, S.F., Shah, M.: High-level event recognition in unconstrained videos. *Int. J. Multimed. Inf. Retr.* **2**(2), 73–101 (2013)
80. Joder, C., Essid, S., Richard, G.: Temporal integration for audio classification with application to musical instrument classification. *IEEE Trans. Audio Speech Lang. Process.* **17**(1), 174–186 (2009). doi:[10.1109/TASL.2008.2007613](https://doi.org/10.1109/TASL.2008.2007613)
81. Jourjine, A., Rickard, S., Yilmaz, O.: Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2985–2988 (2000)
82. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732 (2014)
83. Kay, J.: Feature discovery under contextual supervision using mutual information. In: Proceedings of the International Joint Conference on Neural Networks, vol. 4, pp. 79–84 (1992)
84. Kidron, E., Schechner, Y., Elad, M.: Pixels that sound. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 88–95 (2005)
85. Kijak, E., Gravier, G., Gros, P., Oisel, L., Bimbot, F.: HMM based structuring of tennis videos using visual and audio cues. In: Proceedings of the IEEE International Conference on Multimedia Expo, pp. 309–312. IEEE Computer Society, Washington (2003)
86. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(3), 226–239 (1998)
87. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)
88. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
89. Kuhn, G.F.: Model for the interaural time differences in the azimuthal plane. *J. Acoust. Soc. Am.* **62**(1), 157–167 (1977)
90. Lai, P.L., Fyfe, C.: Kernel and nonlinear canonical correlation analysis. *Int. J. Neural Syst.* **10**(5), 365–378 (2000)
91. Levy, A., Gannot, S., Habets, E.: Multiple-hypothesis extended particle filter for acoustic source localization in reverberant environments. *IEEE Trans. Audio Speech Lang. Process.* **19**(6), 1540–1555 (2011)
92. Li, D., Dimitrova, N., Li, M., Sethi, I.: Multimedia content processing through cross-modal association. In: Proceedings of the ACM International Conference on Multimedia, Berkeley, CA (2003)

93. Lim, A., Nakamura, K., Nakadai, K., Ogata, T., Okuno, H.G.: Audio-visual musical instrument recognition. In: Proceedings of the National Convention Audio-V Information Processing Society (2011)
94. Liu, Q., Wang, W., Jackson, P.J., Barnard, M., Kittler, J., Chambers, J.: Source separation of convolutive and noisy mixtures using audio-visual dictionary learning and probabilistic time-frequency masking. *IEEE Trans. Signal Process.* **61**(22), 5520–5535 (2013)
95. Liutkus, A., Durrieu, J.L., Daudet, L., Richard, G.: An overview of informed audio source separation. In: Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services, pp. 1–4. IEEE, New York (2013)
96. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
97. Mahadevan, V., Li, W., Bhalodia, V., Vasconcelos, N.: Anomaly detection in crowded scenes. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, vol. 249, p. 250 (2010)
98. Makino, S., Lee, T.W., Sawada, H.: Blind Speech Separation. Springer, New York (2007)
99. Mandel, M., Ellis, D.: EM localization and separation using interaural level and phase cues. In: Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 275–278 (2007)
100. Mandel, M., Bressler, S., Shinn-Cunningham, B., Ellis, D.: Evaluating source separation algorithms with reverberant speech. *IEEE Trans. Audio Speech Lang. Process.* **18**(7), 1872–1883 (2010)
101. Maragos, P., Gros, P., Katsamanis, A., Papandreou, G.: Cross-modal integration for performance improving in multimedia: a review. In: *Multimodal Processing and Interaction*, pp. 1–46. Springer, New York (2008)
102. Marti, A., Cobos, M., Lopez, J., Escolano, J.: A steered response power iterative method for high-accuracy acoustic source localization. *J. Acoust. Soc. Am.* **134**(4), 2627–2630 (2013)
103. Metallinou, A., Lee, S., Narayanan, S.: Decision level combination of multiple modalities for recognition and analysis of emotional expression. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2462–2465 (2010)
104. Milani, S., Fontani, M., Bestagini, P., Barni, M., Piva, A., Tagliasacchi, M., Tubaro, S.: An overview on video forensics. *APSIPA Trans. Signal Inf. Process.* **1**, e2 (2012)
105. Monaci, G., Vanderghenst, P.: Audiovisual gestalts. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, pp. 200–200 (2006)
106. Monaci, G., Jost, P., Vanderghenst, P., Mailhé, B., Lesage, S., Gribonval, R.: Learning multimodal dictionaries. *IEEE Trans. Image Process.* **16**(9), 2272–2283 (2007)
107. Monaci, G., Vanderghenst, P., Sommer, F.T.: Learning bimodal structure in audio-visual data. *IEEE Trans. Neural Netw.* **20**(12), 1898–1910 (2009)
108. Moore, B.C.J.: Introduction to the Psychology of Hearing. Macmillan, London (1977)
109. Murphy, K.P.: Dynamic Bayesian networks: representation, inference and learning. Ph.D. thesis, University of California, Berkeley (2002)
110. Naphade, M.R., Garg, A., Huang, T.S.: Audio-visual event detection using duration dependent input output markov models. In: Proceedings of the IEEE Workshop Content-Based Access Image and Video Libraries, pp. 39–43. IEEE, New York (2001)
111. Nefian, A.V., Liang, L., Pi, X., Xiaoxiang, L., Mao, C., Murphy, K.P.: A coupled HMM for audiovisual speech recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2. IEEE, New York (2002)
112. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the International Conference on Machine Learning, pp. 689–696 (2011)
113. Nguyen, V.T., Nguyen, D.L., Tran, M.T., Le, D.D., Duong, D.A., Satoh, S.: Query-adaptive late fusion with neural network for instance search. In: Proceedings of the IEEE International Workshop on Multimedia Signal Processing, pp. 1–6. IEEE, New York (2015)

114. Nikunen, J., Virtanen, T.: Direction of arrival based spatial covariance model for blind sound source separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **22**(3), 727–739 (2014)
115. Omologo, M., Svaizer, P.: Acoustic event localization using a crosspower-spectrum phase based technique. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2. IEEE, New York (1994)
116. Otsuka, T., Ishiguro, K., Sawada, H., Okuno, H.G.: Bayesian nonparametrics for microphone array processing. *IEEE/ACM Trans. Audio Speech Lang. Proc.* **22**(2), 493–504 (2014)
117. Ozerov, A., Févotte, C.: Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation. *IEEE Trans. Audio Speech Lang. Process.* **18**(3), 550–563 (2010)
118. Ozerov, A., Févotte, C., Blouet, R., Durrieu, J.L.: Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Prague (2011)
119. Ozerov, A., Vincent, E., Bimbot, F.: A general flexible framework for the handling of prior information in audio source separation. *IEEE Trans. Audio Speech Lang. Process.* **20**(4), 1118–1133 (2012)
120. Parekh, S., Essid, S., Ozerov, A., Duong, N.Q.K., Pérez, P., Richard, G.: Motion informed audio source separation. In: *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, New Orleans (2017)
121. Parisi, R., Croene, P., Uncini, A.: Particle swarm localization of acoustic sources in the presence of reverberation. In: *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 4. IEEE, New York (2006)
122. Parra, L., Spence, C.: Convolutive blind separation of non-stationary sources. *IEEE Trans. Speech Audio Process.* **8**(3), 320–327 (2000)
123. Pertilä, P., Mieskolainen, M., Hämäläinen, M.: Closed-form self-localization of asynchronous microphone arrays. In: *Proceedings of the Joint Workshop on Hands-free Speech Communication and Microphone Arrays*, pp. 139–144. IEEE, New York (2011)
124. Rocha, A., Scheirer, W., Boulton, T., Goldenstein, S.: Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Comput. Surv.* **43**(4), 26 (2011)
125. Rokach, L.: Ensemble-based classifiers. *Artif. Intell. Rev.* **33**(1), 1–39 (2010). doi:[10.1007/s10462-009-9124-7](https://doi.org/10.1007/s10462-009-9124-7)
126. Roy, R., Kailath, T.: Esprit-estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust. Speech Signal Process.* **37**(7), 984–995 (1989)
127. Sadlier, D.A., O'Connor, N.E.: Event detection in field sports video using audio-visual features and a support vector machine. *IEEE Trans. Circuits Syst. Video Technol.* **15**(10), 1225–1233 (2005)
128. Sawada, H., Mukai, R., Araki, S., Makino, S.: A robust and precise method for solving the permutation problem of frequency-domain blind source separation. *IEEE Trans. Speech Audio Process.* **12**(5), 530–538 (2004)
129. Schau, H., Robinson, A.: Passive source localization employing intersecting spherical surfaces from time-of-arrival differences. *IEEE Trans. Acoust. Speech Signal Process.* **35**(8), 1223–1225 (1987)
130. Scheuing, J., Yang, B.: Disambiguation of tdoa estimation for multiple sources in reverberant environments. *IEEE Trans. Audio Speech Lang. Process.* **16**(8), 1479–1489 (2008)
131. Schmidt, R.: Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas Propag.* **34**(3), 276–280 (1986)
132. Sedighin, F., Babaie-Zadeh, M., Rivet, B., Jutten, C.: Two multimodal approaches for single microphone source separation. In: *Proceedings of the European Signal Processing Conference* (2016)
133. Seichepine, N., Essid, S., Févotte, C., Cappe, O.: Soft nonnegative matrix co-factorization with application to multimodal speaker diarization. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver (2013)
134. Seichepine, N., Essid, S., Févotte, C., Cappe, O.: Soft nonnegative matrix co-factorization. *IEEE Trans. Signal Process.* **PP**(99) (2014)

135. Serizel, R., Moonen, M., van Dijk, B., Wouters, J.: Low-rank approximation based multi-channel wiener filter algorithms for noise reduction with application in cochlear implants. *IEEE/ACM Trans. Audio Speech Lang. Process.* **22**(4), 785–799 (2014)
136. Serizel, R., Bisot, V., Essid, S., Richard, G.: Machine listening techniques as a complement to video image analysis in forensics. In: *Proceedings of the IEEE International Conference on Image Processing*, pp. 948–952. IEEE, New York (2016)
137. Showen, R., Calhoun, R., Dunham, J.: Acoustic location of gunshots using combined angle of arrival and time of arrival measurements (2009). US Patent 7,474,589
138. Sigg, C., Fischer, B., Ommer, B., Roth, V., Buhmann, J.: Nonnegative CCA for audiovisual source separation. In: *Proceedings of the IEEE Workshop Machine Learning and Signal Processing*, pp. 253–258. IEEE, New York (2007)
139. Smaragdis, P., Casey, M.: Audio visual independent components. In: *Proceedings of the International Symposium Independent Component Analysis and Blind Signal Separation*, pp. 709–714 (2003)
140. Srivastava, N., Salakhutdinov, R.R.: Multimodal learning with deep Boltzmann machines. In: *Proceedings of the Advances in Neural Information Processing Systems*, pp. 2222–2230 (2012)
141. P. Stoica, Moses, R.: *Spectral Analysis of Signals*. Pearson Prentice Hall, Upper Saddle River, NJ (2005)
142. Strobel, N., Spors, S., Rabenstein, R.: Joint audio-video object localization and tracking. *IEEE Signal Process. Mag.* **18**(1), 22–31 (2001)
143. Tian, Y., Chen, Z., Yin, F.: Distributed Kalman filter-based speaker tracking in microphone array networks. *Appl. Acoust.* **89**, 71–77 (2015)
144. Togami, M., Hori, K.: Multichannel semi-blind source separation via local Gaussian modeling for acoustic echo reduction. In: *Proceedings of the European Signal Processing Conference* (2011)
145. Togami, M., Kawaguchi, Y.: Simultaneous optimization of acoustic echo reduction, speech dereverberation, and noise reduction against mutual interference. *IEEE/ACM Trans. Audio Speech Lang. Process.* **22**(11), 1612–1623 (2014)
146. Trifa, V., Koene, A., Moren, J., Cheng, G.: Real-time acoustic source localization in noisy environments for human-robot multimodal interaction. In: *Proceedings of the IEEE International Symposium on Robots and Human Interactive Communication* (2007)
147. Valente, S., Tagliasacchi, M., Antonacci, F., Bestagini, P., Sarti, A., Tubaro, S.: Geometric calibration of distributed microphone arrays from acoustic source correspondences. In: *Proceedings of the IEEE International Workshop on Multimedia Signal Processing*, pp. 13–18 (2010)
148. Valin, J., Michaud, F., Rouat, J.: Robust 3d localization and tracking of sound sources using beamforming and particle filtering. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4. IEEE, New York (2006)
149. Velivelli, A., Ngo, C.W., Huang, T.S.: Detection of documentary scene changes by audio-visual fusion. In: *Proceedings of the International Conference on Image and Video Retrieval*, pp. 227–238. Springer, New York (2003)
150. Vincent, E., Bertin, N., Gribonval, R., Bimbot, F.: From blind to guided audio source separation: how models and side information can improve the separation of sound. *IEEE Signal Process. Mag.* **31**(3), 107–115 (2014)
151. Vuegen, L., Broeck, B.V.D., Karsmakers, P., hamme, H.V., Vanrumste, B.: Automatic monitoring of activities of daily living based on real-life acoustic sensor data: a preliminary study. In: *Proceedings of the International Workshop on Speech and Language Processing for Assistive Technologies*, pp. 113–118 (2013)
152. Wang, D.L.: Time-frequency masking for speech separation and its potential for hearing aid design. *Trends Amplif.* **12**(4), 332–352 (2008)
153. Wang, H., Chu, P.: Voice source localization for automatic camera pointing system in videoconferencing. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (1997)

154. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. *Int. J. Comput. Vis.* **103**(1), 60–79 (2013)
155. Ward, D.B., Lehmann, E.A., Williamson, R.C.: Particle filtering algorithms for tracking an acoustic source in a reverberant environment. *IEEE Trans. Speech Audio Process.* **11**(6), 826–836 (2003)
156. Wilkins, P., Adamek, T., Byrne, D., Jones, G., Lee, H., Keenan, G., Mcguinness, K., O'Connor, N.E., Smeaton, A.F., Amin, A., Obrenovic, Z., Benmokhtar, R., Galmar, E., Huet, B., Essid, S., Landais, R., Vallet, F., Papadopoulos, G.T., Vrochidis, S., Mezaris, V., Kompatsiaris, I., Spyrou, E., Avrithis, Y., Morzinger, R., Schallauer, P., Bailer, W., Piatrik, T., Chandramouli, K., Izquierdo, E., Haller, M., Goldmann, L., Samour, A., Cobet, A., Sikora, T., Praks, P.: K-space at TRECVID 2007. In: *TRECVID 2007* (2007)
157. Wu, Y., Lin, C.Y.Y., Chang, E.Y., Smith, J.R.: Multimodal information fusion for video concept detection. In: *Proceedings of the IEEE International Conference on Image Processing*, vol. 4, pp. 2391–2394. IEEE, Singapore (2004)
158. Wu, Z., Jiang, Y.G., Wang, J., Pu, J., Xue, X.: Exploring inter-feature and inter-class relationships with deep neural networks for video classification. In: *Proceedings of the ACM International Conference on Multimedia*, pp. 167–176. ACM, New York (2014)
159. Yilmaz, K., Cemgil, A.T.: Probabilistic latent tensor factorisation. In: *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation*, pp. 346–353 (2010)
160. Yokoya, N., Yairi, T., Iwasaki, A.: Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion. *IEEE Trans. Geosci. Remote Sens.* **50**(2), 528–537 (2012)
161. Yoo, J., Choi, S.: Matrix co-factorization on compressed sensing. In: *Proceedings of the International Joint Conference on Artificial Intelligence* (2011)
162. Yost, W.A.: Discriminations of interaural phase differences. *J. Acoust. Soc. Am.* **55**(6), 1299–1303 (1974)
163. Yuhas, B.P., Goldstein, M.H., Sejnowski, T.J.: Integration of acoustic and visual speech signals using neural networks. *IEEE Commun. Mag.* **27**(11), 65–71 (1989)
164. Zhang, Q., Chen, Z., Yin, F.: Distributed marginalized auxiliary particle filter for speaker tracking in distributed microphone networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(11), 1921–1934 (2016)
165. Zotkin, D.N., Duraiswami, R.: Accelerated speech source localization via a hierarchical search of steered response power. *IEEE Trans. Speech Audio Process.* **12**(5), 499–508 (2004)

Part IV

Applications

Chapter 10

Sound Sharing and Retrieval

Frederic Font, Gerard Roma, and Xavier Serra

Abstract Multimedia sharing has experienced an enormous growth in recent years, and sound sharing has not been an exception. Nowadays one can find online sound sharing sites in which users can search, browse, and contribute large amounts of audio content such as sound effects, field and urban recordings, music tracks, and music samples. This poses many challenges to enable search, discovery, and ultimately reuse of this content. In this chapter we give an overview of different ways to approach such challenges. We describe how to build an audio database by outlining different aspects to be taken into account. We discuss metadata-based descriptions of audio content and different searching and browsing techniques that can be used to navigate the database. In addition to metadata, we show sound retrieval techniques based on the extraction of audio features from (possibly) unannotated audio. We end the chapter by discussing advanced approaches to sound retrieval and by drawing some conclusions about present and future of sound sharing and retrieval. In addition to our explanations, we provide code examples that illustrate some of the concepts discussed.

Keywords Sound sharing • Sound retrieval • Multimedia • Audio metadata • Sound description • Audio database • Audio indexing • Audio features • Similarity search • Query by example • Sound taxonomy • Machine learning • Sound exploration • Sound search

10.1 Introduction

Multimedia sharing is one of the areas in which the social web has experienced the largest and quickest growth in recent years [73]. Just to name a few examples, every minute 100h of video are uploaded to YouTube [75], 2400 photos are uploaded

F. Font (✉) • X. Serra
Music Technology Group (MTG), Universitat Pompeu Fabra, Barcelona, Spain
e-mail: frederic.font@upf.edu; xavier.serra@upf.edu

G. Roma
Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK
e-mail: g.roma@surrey.ac.uk

to Flickr [32], and 12 h of music are uploaded to SoundCloud [79]. The case of *sound* sharing—understanding sound as any kind of reusable audio material like sound effects, environmental recordings, or music building blocks, but typically not finished music tracks—even if at a smaller scale, is not an exception. Websites such as Freesound, Looperman, CC-mixer, and the Radio Aporee project¹ are examples of sound sharing platforms in which users can not only search and browse content but also contribute new audio recordings including sound scenes such as field and urban recordings, sound events such as *foley* or animal sounds, and music samples such as loops, melodies, and single notes. Furthermore, consumer-oriented websites such as Sound Dogs, Sound Snap, and A Sound Effect² sell sounds from extensive collections of audio recordings that users can also navigate. In this chapter we focus on the particular context of online sound sharing and give an overview of different ways to approach sound sharing and retrieval challenges.

On a prototypical scenario of online sound sharing, a user might record a sound and upload it to a web application so that other users can listen to it and possibly download it. The intent with which users upload and share multimedia content can vary widely, but we can identify some general patterns according to the usage that the uploaders may expect of the contributed content. On the one hand, we can identify content that is meant to be accessed and viewed or played through the online sharing platform itself. Hence, the *end use* of the resource is its online consumption. For example, someone may upload photos of an event to a photo sharing site so that other participants of that event can have access to the photos, or a musical artist can upload a music album to a music sharing site so that other users can listen to it. On the other hand, there is an additional type of uploaded content which is meant to be reused outside the sharing platform where it is hosted. Here, the display in the sharing platform does not represent an end use per se. Some examples of this situation include sharing recordings of sound events that can be later used in video games, drum loops in music compositions, video backgrounds or transitions to be used in audiovisual installations, or images to be used in collages or as a desktop wallpaper. These latter cases of multimedia sharing particularly support Lawrence Lessing's definition of *read/write culture* [39]. In read/write culture, users are both consumers and producers of content that is easily shared and reused through the internet [80].

Such content potentially represents an incredibly valuable resource that can serve several purposes, ranging from business and research applications to artistic creation and the preservation of cultural heritage [34]. Nevertheless, the value of this content is significantly dimmed by the ways in which it can be accessed and reused, i.e., the ways in which it can be retrieved. As the amount of content grows, so does the difficulty of browsing and locating what one needs, and so do the challenges that search engines have to face. For the content to be accessible, it needs to be properly indexed. However, the quantity and variety of available content turns

¹<https://freesound.org>, <https://looperman.com>, <http://ccmixter.org>, <http://aporee.org/maps>.

²<https://sounddogs.com>, <https://soundsnap.com>, <https://asoundeffect.com>.

proper indexing into a very difficult task. This is particularly true for multimedia resources like video, pictures, and audio which, as opposed to other kinds of media, do not have a direct textual representation [5]. At the same time, the amount of content generated is simply too much to be curated in scalable ways by groups of experts.

The description, indexing, and retrieval of audio content is therefore a challenge that needs to be faced in order to make audio shareable and increase its value. Especially in the context of read/write culture, users need sophisticated and specialized ways of accessing online resources that fit their particular requirements. Users searching for content in sound sharing sites might be looking for audio clips with very specific and detailed characteristics that can be represented by a wide range of audio properties. For example, one user might be searching for the sound of an opening door with a particular duration, size, and material of the door, while another user might be searching for the sound of a melody being played by a particular instrument with a specific tonality, tempo, and mood. Being able to successfully retrieve such specific content poses a number of issues to both the users and the sharing platform. Another relevant aspect of sound sharing is that the assessment of the results returned by a search engine of a sound sharing site requires the time to listen to them, and cannot be done as instantly as it could be done with the search results of, for example, a photo sharing site. From this point of view, the cost of iterating over several queries in order to find the desired resource is higher for sounds than for images. This is one of the reasons why good quality descriptions are crucial for indexing audio.

But how should sounds be described so that users can effectively search them? As it might be expected, this question does not have a single definitive answer. Nevertheless, we can intuitively differentiate at least two ways in which sounds can be described. On the one hand, sounds can be generally described by referring to the source that produces them. In other words, we can describe a sound by denoting an object and (possibly) an action that produces it (e.g., “the sound of a closing door”). On the other hand, sounds can be described by referring to their perceptual qualities regardless of their source, that is to say, by describing the timbre and acoustic qualities of a perceived sound (e.g., “a loud high-pitched sound”). Both approaches are complementary and both bear relevant information for indexing and retrieval purposes [43]. Source-based descriptions can be effectively indexed by using metadata annotations such as labels and textual descriptions. Conversely, some perceptual qualities can be better represented using automatically extracted audio features. In addition, other sound properties such as audio format and editorial information can be used when indexing content. Once content is described and indexed, different browsing and searching strategies can be implemented such as text-based search, sound browsing based on category filtering, or search based on audio similarity. All these strategies and many other possible ones ultimately enable sound search and discovery.

In this chapter we go through the different components and the typical issues and solutions of sound sharing systems, and show step by step how to build a basic system with references to code examples. The components we describe are similar

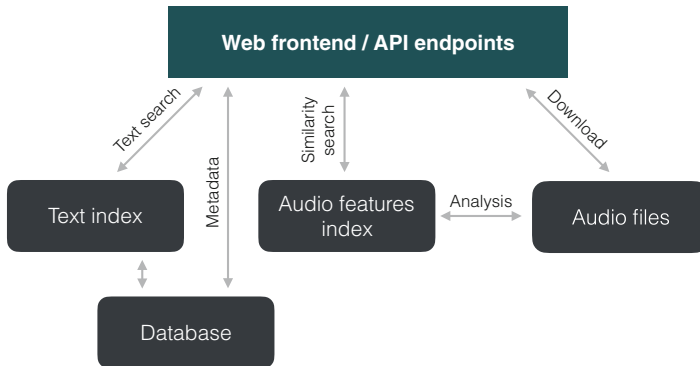


Fig. 10.1 Block diagram of Freesound’s main components. The *arrows* correspond to main functionalities of Freesound and show the components that they need to access. Note that metadata information and audio features are indexed separately and provide different functionalities. Also note that audio files are not included in the database itself (see Sect. 10.2). This diagram serves as a general example of the different components of a sound sharing and retrieval system

to those that can be found in a site such as Freesound, which we take as an example (Fig. 10.1). We start by describing how to build an audio database (Sect. 10.2), and continue by explaining metadata-based retrieval strategies (Sect. 10.3) as well as other retrieval strategies based on audio analysis (Sect. 10.4). We end this chapter with a discussion about advanced audio retrieval topics (Sect. 10.5) and a conclusions section (Sec. 10.6). Code examples, written in the Python³ programming language and based on open source technologies, are available in the book’s accompanying website⁴ and demonstrate some of the concepts discussed in Sects. 10.2–10.4.

10.2 Database Creation

The concept of a database is traditionally associated with text and numerical information. While many database programs can store binary objects, standard practice for applications involving image, audio, or video files is to leave them in the file system and store the paths in the database. The expression *audio database* has been used during the last decades in research on audio analysis to refer to collections of audio files (e.g., [20, 23, 25]). Much of this research was precisely trying to find a way to index audio files and facilitate search and discovery. In this chapter, we will refer to an audio database as the set of information used for indexing collections of

³<https://python.org>.

⁴www.TODO:bookwebsite.

audio files. In the following subsections, we summarize the most important design issues for designing an audio database. First we discuss practical aspects such as file formats and licensing, then we highlight the key design issues related with the two main types of information used for indexing audio: metadata and audio features.

The code examples provided in the accompanying website of this book show how to build an audio database taking into account the design issues discussed below. We show how to download a small number of sounds from Freesound in a standardized preview format using the Freesound API.⁵ We retrieve sounds that match the keywords *dog bark*, *cat meow*, *lion roar*, and *nightingale*, and also retrieve its associated metadata and some pre-computed audio features. Then we show how to store this information in a data structure and create a text index using popular Python libraries. Both data structures are used in later sections of this chapter to demonstrate metadata-based and audio-based retrieval strategies (Sects. 10.3 and 10.4, respectively).

10.2.1 *Licensing, File Formats, and Size*

Licensing terms for audio are more of a legal than a technical issue. However, they will often be a key factor for anyone building an audio database, especially for shared usage. Copyright licensing will be generally needed for audio recordings but additional licensing may be required. For example, if the performance of a musical composition appears in the recording, a license for the composition also applies. There are two interactions to consider: first, the database developer or administrator needs a license from the content author. Second, database users willing to play or download audio files for reuse need a license too. In many social media applications (such as Freesound or Flickr) a license is chosen by the content author at upload time and is propagated by the database to the end users. The development of Creative Commons⁶ (CC) licenses was instrumental in the emergence of this use case because these allow re-distribution under clear terms. Re-distribution is generally not permitted under traditional copyright. While their applicability depends on the audio material (e.g., whether existing protected content is considered) and/or possible commercial agreements, CC licenses provide mature legal terms curated by copyright law experts that may be useful to developers with little or no expertise on this subject.

File format can be naively seen as a trivial issue. Even though it may be trivial for a database designed for individual usage, for collective usage it is convenient to support multiple formats. Choice is complicated due to the plethora of available options, each with its own implications, often due to commercial factors. For this reason, early planning is advised. As a general rule, variety of formats is convenient

⁵https://freesound.org/docs/api/resources_apiv2.html.

⁶<https://creativecommons.org>.

for collecting sounds from different authors (who may use different equipment), and uniformity is convenient to facilitate reuse. An obvious general way to classify formats is to distinguish between *uncompressed*, *lossy* (i.e., when the compression implies loss of information), and *lossless* compressed formats. In some cases, a distinction between *container* and *codec* may be relevant, especially when dealing with video files. In pure audio formats there is often no distinction between container and codec, so we will not make that distinction in this chapter either. In addition to the format, there are numerous possible combinations of sampling rate and sample bit depth. Audio production textbooks are a good source for conventional choices of sampling rate and bit depth [29]. The most universal is the one defined by the compact disc standard, 44.1 kHz and 16 bits. With respect to uncompressed formats, WAV and AIFF are the most common. As an example of non-trivial issue, 24-bit integers are a very common choice, yet some high-level languages such as Python lack a specific 24-bit primitive data type, and as a consequence reading 24-bit wav files in some Python packages will not work. Similar situations may be encountered with 32-bit audio and for any obscure combinations: libraries for dealing with different audio formats may surprisingly fail to recognize many files. Lossy compression has been instrumental in the popularization of digital audio in the Internet age. Perhaps the main issue is that many formats are covered by patents. At the time of this writing, MP3 patents are progressively expiring. Open formats such as OGG are free to use but often not supported in commercial players and devices. A similar situation is found with respect to lossless compressed formats: while open formats such as FLAC are available, companies often develop their own formats and protect them, so they are mostly used in specific platforms. Compressed formats will introduce more possibilities for variation, typically a bit rate or quality parameter. A common sense strategy is to allow authors to contribute content with the format they want, and then convert to a standardized format. For large scales, if an uncompressed format is chosen as standard and/or the originals are preserved, this approach may require large amounts of storage space.

File size and duration is also something to consider when creating an audio database, not only in order to plan the storage needs but also for deciding the way in which files are accessed. Short audio recordings will usually contain sound events, while longer recordings may contain music, speeches, or environmental sound scenes. Analysis and segmentation of longer recordings may be of interest for some applications in order to isolate specific events or to allow streaming. Large file sizes also complicate transmission for authors, for example, when using HTTP to upload files to a database server.

10.2.2 Metadata

Metadata can be defined as “data about data.” In the case of audio it usually refers to textual information that is used to describe and index an audio file or segment. Virtually all existing sound sharing platforms implement some kind of metadata-

based retrieval strategy. Text is the most established way to deal with any sort of information stored in computers, so for most applications, some kind of metadata is necessary. Audio files already contain some sort of metadata in the headers, such as sampling rate, bit depth, bit rate, and potentially editorial information, that can be added to the database for indexing. Nevertheless, sound sharing platforms often delegate the responsibility of providing metadata to the content authors or editors. This will typically include a name (which may or may not coincide with the file name), a textual description of the content of the sound, a number of labels or *tags*, or other more structured bits of information such as audio file format properties, time of recording, or geo-location information. Even though audio features could also be considered to be metadata, these are typically excluded from the definition.

A general concern is the consistency of the provided metadata, which is affected by the original design of the data model. Like in the case of file formats, some degree of freedom in terms of required metadata will allow to make an audio database more attractive to different users. For example, tags have become a very popular way to attach text labels to pieces of information without any predefined structure. Conversely, more structured and strict metadata layouts may benefit indexing and retrieval in some cases. In Sect. 10.3.1 a more detailed discussion is given regarding metadata fields and consistency.

In order to index and retrieve content based on text metadata, a full-text search engine is especially useful. Full-text search engines are specialized software programs that are useful for searching in text documents or textual representations of documents as metadata fields. The choice of indexing algorithms depends on the type of information stored in a database and the way it is to be retrieved. The availability of implementations will typically determine the choice of a given database program or library. Traditional relational databases typically rely on B+ trees for indexing. Other tree structures can be used for spatial queries as described in Sect. 10.4. While the plethora of available database software is beyond the scope of this chapter, it is useful to distinguish three main groups that are commonly helpful for large-scale applications. Traditional relational databases, also known as relational database management systems (RDBMS), are used via the structured query language (SQL) in many corporate and web applications. The more modern trend of *NoSQL* databases comprises a heterogeneous group including document databases, key-value stores, and graph databases. Some of these use text formats commonly used for audio metadata, such as XML or JSON. For information indexed using complex ontologies, specialized graph databases or triple-stores may be needed (see Sect. 10.5) [59].

10.2.3 Audio Features

Audio features or *descriptors* are numerical representations obtained through automatic analysis of audio, often attempting to capture some aspect of human perception. A large number of such descriptors have been developed over the

years through research in automatic speech recognition (ASR), music information retrieval (MIR), and environmental sound recognition (ESR), including sound scene and sound event analysis (see Chap. 4). The use of audio features has been historically different in each of these domains. For speech, features were used most of the time as a spectral representation of sound used to train discrete models (e.g., hidden Markov models) informed by human speech and language. For music, features have been traditionally organized in different levels, according to their proximity to music theory concepts. For environmental sound most of the time very generic features are used in order to recognize specific semantic categories. Creating a database of audio features involves feature extraction software. One example can be found in the Freesound Extractor,⁷ which extracts a number of audio features using the Essentia [8] audio analysis library. The most important question when designing a database using audio features is to know which features are relevant to the expected type of content and use case. So far there are mainly two interaction paradigms that have been extensively researched for audio retrieval based on features: range queries can be used with descriptors that are understandable for humans. A simple example is finding pitched sounds within a given range of pitches. Query by example (often also called *similarity search*) refers to using an example sound to find similar sounds in the database (see Sect. 10.4). In the field of data-driven music creation a special case is to find sequences of shorter sounds in the database that are optimally close to an audio query. This technique has been named *musical mosaicing* [81] or *concatenative sound synthesis* [67].

10.3 Metadata-Based Sound Retrieval

Metadata is the most common way through which audio databases can be navigated and their content retrieved. In this section we describe in more detail the use of user-provided metadata for indexing audio content (Sect. 10.3.1) and explain some of the most common sound retrieval strategies based on metadata (Sect. 10.3.2). The code examples referenced along with the explanations build up from the examples referenced in the previous section.

10.3.1 Metadata for Audio Content

Content authors or editors typically provide metadata in the form of a number of *annotations* or *descriptions*. As described in Sect. 10.2, it is common for sound sharing platforms to rely on such user-provided metadata for audio indexing and retrieval. Nevertheless, the nature of content annotations may vary on each particular

⁷<https://github.com/MTG/essentia/tree/master/src/examples/freesound>.

sharing platform, and is highly dependent on the description strategy used in every particular site. Description strategies that look for the most uniform annotations can use forms with a number of predefined metadata fields with fixed responses. For each field, users will choose one of the available responses when describing a resource. For example, users might be asked to classify a sound effect by selecting a category from a fixed list of categories. However, these strategies lack flexibility when new resources are uploaded because their characteristics can be unexpected and not contemplated in the description form [27, 42, 71]. Other description strategies provide more flexibility by not limiting metadata fields to a specific set of responses. In that case, annotations typically consist of a textual description and a number of tags which are not restricted to a particular vocabulary.

Using tags as keywords for annotating resources has become standard practice in many online sharing sites of very different nature. Just to name a few examples, multimedia sharing sites like YouTube, Vimeo, Flickr, SoundCloud, Bandcamp, Last.fm, or Freesound⁸ have content labeled using tags. However, despite the popularity of tagging systems and their successful implementation in many online sharing sites, there are a number of well-known problems which limit the possibilities of these functionalities [26]. These problems range from the use of different tags to refer to a single concept (synonymy) and the ambiguity in the meaning of certain tags (polysemy), to tag scarcity and typographical errors [24, 27]. Furthermore, the quality of the indexing, searching, and browsing functionalities enabled by tagging systems strongly relies on the coherence and comprehensiveness of the tags assigned to the resources. It is not only important that individual resources are properly tagged, but also that descriptions are consistent across the database. For that reason, it has been often discussed whether a tagging system, after a certain time of being in use, reaches a point of implicit consensus where the vocabulary converges to a certain set of tags and tagging conventions that are widely adopted by all users of the system [27, 58, 70, 74, 78]. According to these authors, the point of consensus may be reached because of imitation patterns and users' shared cultural knowledge. Reaching that point of consensus is desirable to improve indexing and overall sharing experience [26]. It is a common strategy in sharing platforms to use tag recommendation methods to help users during the description process [19, 24, 27, 44]. By using such methods, user annotations are expected to be more uniform and comprehensive, thus helping in reaching the aforementioned point of consensus.

Overall, the choice of using a flexible description strategy or a more strict one strongly depends on the nature of the data that needs to be collected. Flexible systems are a better fit for heterogeneous data, while strict systems can work well for cases in which the information to annotate is very well defined. In the case of audio material a mixed approach can be a good option, using well-defined metadata fields relating to aspects such as audio format (e.g., sample-rate, bit depth, number

⁸<https://youtube.com>, <https://vimeo.com>, <https://flickr.com>, <https://soundcloud.com>, <https://bandcamp.com>, <https://last.fm>, <https://freesound.org>.

of channels) or other recording properties (e.g., duration, date of recording, duration of recording, used microphone, etc.), and also using more flexible fields such as a textual description of the activities being recorded. In the specific case of sound events and sound scenes, it is important to put an emphasis on describing the sound sources that are captured in a recording (i.e., *what* produces the sound). Because of the potential variety of sound sources, tagging systems are typically appropriate for annotating that kind of information. For sound scenes, it is also desirable to attach annotations to particular regions of a recording, being able in this way to provide specific descriptions for different fragments of the scene.

10.3.2 Search and Discovery of Indexed Content

As previously mentioned, a common choice for indexing metadata is the use of a full-text search engine. In that case users typically introduce some search terms (i.e., words) as a query. The search engine then matches these terms with indexed metadata fields and returns a sorted list of results (sounds in our examples). For each sound in the index, the search engine will compute a relevance score based on how well the input terms match the information in the metadata fields and how relevant the matched terms are. The classic relevance score in information retrieval is based on calculating the relevance of a term with respect to a given document by the TF*IDF measure [72]. TF stands for “term frequency,” and number of times the specific term appears in a document. IDF stands for “inverse document frequency,” and represents the inverse of the number of documents in the index that contain the given term. The idea is that a given term will be relevant with respect to a document if it appears many times, but the relevance will be penalized if it also appears in many other documents. Using such a relevance function and given a number of input query terms, a global score can be computed by aggregating the relevance of each query term for the different metadata fields of each document in the index. Other common score functions include the BM25F, which is a variation of TF*IDF based on probabilistic information retrieval [57], and the PageRank algorithm [50], which calculates the relevance of a document based on the relevance of documents that point to it.

Besides the scoring functions for sorting results, search engines can include *query expansion* mechanisms which perform pre-processing of user queries before matching it with the index contents [14]. The idea behind query expansion is that the input terms provided by a user can be expanded with other relevant terms before matching with the index, potentially increasing the number of results. The way in which new terms are added to the query can be based on simple strategies such as using synonym lists or on more complex strategies such as the analysis of previous queries or the use of domain-specific knowledge (see Sect. 10.5). Search results may be further refined by allowing users to specify filtering criteria. In this way metadata fields that are not taken into account in the scoring function can be used in the search process to restrict the searchable space.

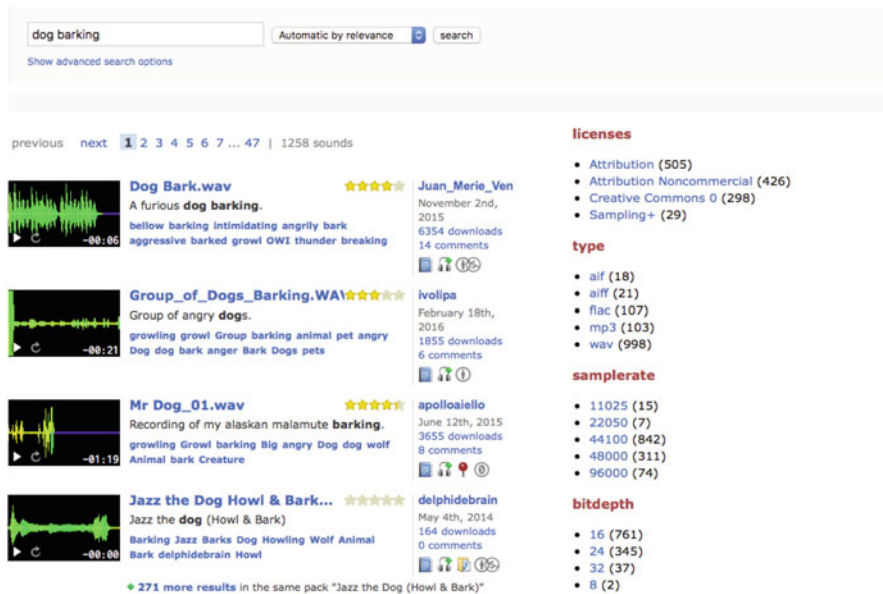


Fig. 10.2 Example of faceted search as implemented in Freesound. Facets for *licenses*, *type*, *samplerate*, and *bitdepth* metadata fields are displayed for the results of the query “dog barking.” Clicking on these facets further filters the query results

Another typical retrieval strategy based on metadata is the use of *facets* when displaying search results. This is typically referred to as *faceted search* [76]. Given the results of a query, faceted search extends conventional search by dynamically summarizing the distribution of values in a number of information facets (i.e., metadata fields) and showing this information to users. In this way, users can use the information displayed in facets to further filter and update their queries (see Fig. 10.2). Faceted search has become increasingly popular in sharing platforms and provides a foundation for interactive information retrieval by allowing iterative results-informed query refinement.

Faceted search allows the discovery of the database beyond conventional search by providing users with a way to *navigate* the content (even without specifying initial query terms). A particularly successful faceted search application is the use of a *tag cloud* as a browsing interface. A tag cloud shows the most commonly used tags in a database with the size of each tag set proportional to its frequency of occurrence (Fig. 10.3) [33]. Users can typically navigate a collection by applying query filters based on the tags in the tag cloud, and for each new filter a new tag cloud can be computed and displayed only considering the filtered set of documents. Note that when new content is indexed in a database, the tag cloud can be automatically updated. Therefore tag clouds show an up to date overview of the contents of a database.

ambiance⁺ ambience⁺ ambient⁺ atmosphere⁺ background⁺ bass⁺ beat⁺ bell⁺
 bird⁺ birds⁺ car⁺ cars⁺ city⁺ click⁺ close⁺ computer⁺ creepy⁺ crowd⁺ dance⁺ dark⁺ deep⁺ digital⁺
 door⁺ drone⁺ drum⁺ drums⁺ effect⁺ electric⁺ electro⁺ electronic⁺ engine⁺ english⁺
 female⁺ field⁺ field-recording⁺ foley⁺ footsteps⁺ forest⁺ fx⁺ game⁺ girl⁺ glass⁺ glitch⁺
 guitar⁺ hit⁺ horror⁺ house⁺ human⁺ industrial⁺ kick⁺ kitchen⁺ loop⁺ machine⁺ male⁺
 metal⁺ motor⁺ multisample⁺ music⁺ nature⁺ night⁺ noise⁺ open⁺ owl⁺ people⁺
 percussion⁺ piano⁺ processed⁺ radio⁺ rain⁺ recording⁺ reverb⁺ rhythm⁺ sample⁺ scary⁺ sci-fi⁺
 sfx⁺ sound⁺ soundscape⁺ space⁺ speak⁺ speech⁺ stereo⁺ storm⁺ street⁺ synth⁺ synthesizer⁺
 talk⁺ techno⁺ thunder⁺ traffic⁺ train⁺ vocal⁺ voice⁺ walking⁺ water⁺ weather⁺ weird⁺
 wind⁺ woman⁺ wood⁺

Fig. 10.3 Example tagcloud taken from Freesound (retrieved September 27th 2016)

Our provided code examples show a simple implementation of a number of metadata-based sound retrieval strategies. We provide a basic text search system which is configured to match input query terms with the information in *name*, *description*, and *tags* metadata fields. We also provide examples for filtering search results based on *duration* and sound *license* fields. Furthermore, our example code shows how to define facets and group search results based on these (again using *license* and *duration* metadata fields). Finally, we provide code to generate a tagcloud which summarizes the contents of an audio database by displaying its most important tags.

10.4 Audio-Based Sound Retrieval

Audio-based retrieval (also known as content-based retrieval) refers to the use of descriptors computed automatically from the waveform in order to find audio files in a database. The obvious advantage of using automatic descriptors is that it allows indexing content when no labels or metadata are available. Manual labelling may require significant amounts of work, which could be avoided by using automatic analysis. On the other hand, currently available descriptors do not always bear an intuitive meaning for non-expert (or even expert) users. Often they are used in conjunction with machine learning algorithms in order to obtain meaningful labels (i.e., typically through segmentation and/or classification). In this section we will review basic retrieval techniques using audio descriptors. More details on specific descriptors can be found in Chap. 4.

10.4.1 Audio Features

One of the most critical aspects for audio-based retrieval is devising a set of features that are relevant for the application. The standard representation of audio as a waveform in the time domain allows for visual inspection and graphical editing, but conveys only limited information (i.e., amplitude) about the sound. Since time-frequency transforms such as the Short-time Fourier Transform (STFT) became affordable and ubiquitous in audio processing, the spectrogram has been used as a more intuitive representation. Most descriptors used for content-based indexing of audio data are related to some time-frequency transform, and attempt to describe some aspect of it that is relevant to some application or is intuitively useful. An obvious example would be detecting the pitch for harmonic signals. A large number of software libraries are available for feature extraction, mainly in the context of MIR research [8, 10, 11, 36, 45, 46, 54, 77]. It is common in MIR to distinguish low-level features (i.e., closer to the spectral representation but with little intuitive meaning) and higher-level features related with musical concepts. Environmental audio can be seen as a very general case where it is still possible to find music (e.g., street music, the radio in a car) and very likely pitched sounds such as human or other animal vocalizations, or human-originated sounds like alarms. Finding a generic representation is not straightforward and in most cases it may depend on the application. Low-level features can still be used for most kinds of sounds. A popular set of low-level features was compiled in the definition of the MPEG-7 standard [55]. As generic descriptors, Mel-frequency cepstral coefficients (MFCCs) are still very widely used, like in the case of music and speech. More recently, deep learning architectures make it possible to automatically learn the required representations from spectral frames [38], or even from raw audio waveforms [30]. A strategy for supporting many potential applications is computing a comprehensive set of features. This is the strategy adopted in the aforementioned Freesound Extractor (Sect. 10.2.3), and does not bear a large computational cost in respect to the whole operation.

10.4.2 Feature Space

Audio features are typically aggregated along time in order to represent each audio file as a single vector. This process may depend on the content of the file: for sound events, the temporal evolution of spectral features may be taken into account, while for sound scenes global statistics may suffice. Statistics can also be computed from delta features, producing higher-dimensional vectors that capture some of the short-term temporal evolution. The idea of ignoring the actual order of spectral features and computing statistics has been dubbed the “bag of frames” approach [3], by analogy to the “bag of words” model in text retrieval. Feature vectors can also be extracted from events detected in long recordings.

The set of selected features can then be seen to form a vector space, defined by some distance metric between feature vectors. Typical choices are distance metrics associated with p -norms, or cosine distance for large dimensions. Feature vectors represent audio documents, and can be added as records to any conventional database software. On the other hand, it is very convenient if the collection can be kept in computer memory. As an example, for an arbitrary distance metric, a naive query-by-example approach would require traversing all the vectors and compute the distance to the query vector (see Sect. 10.4.4). Since there may be some redundancy in the feature space (for example, when a very large set of features is computed), dimensionality reduction algorithms (e.g., principal component analysis) can be helpful. Another common practice is to quantize features into typically sparse spaces, e.g., using data clustering [37, 64]. However, keeping the original features allows defining distance metrics over meaningful subsets. This is especially useful if many different kinds of sounds are mixed, so different feature sets can be used by different applications.

10.4.3 *Descriptor-Based Queries*

Using descriptors directly in user interfaces is not very common, especially when dealing with general audio, since low-level features often lack an intuitive meaning. Moreover, the distribution of features across the database must be taken into account in order to find sensible values.⁹ Queries based on hand-picked descriptor targets or ranges are still possible for some low-level features. An intuitive example is a rough division of the power spectrum in a few frequency bands (as is used frequently in audio production and mixing, e.g., *low*, *mid-low*, *mid-high*, *high*). A non-specialist user could use these bands to select sounds where most of the energy is in the higher frequencies (e.g., selecting bird sounds), or in the lower frequencies (e.g., to find sounds of passing cars). Descriptor-based queries can also be devised by an expert and presented as discrete choices to the user. An even simpler example is using the spectral centroid (see the code examples provided in the book website). Since audio descriptors are often floating point numbers, a common strategy is to specify a query range for a given descriptor or for a set of them. However, more complex queries can be made using specialized languages like SQL or other database query languages. In concatenative sound synthesis research, it was also common to use audio features as axes of interactive scatterplots of audio collections [68]. The exploration can also be driven by gestures [17].

⁹For this reason, histograms are provided as part of the documentation of the Freesound API: https://www.freesound.org/docs/api/analysis_docs.html.

10.4.4 *Query by Example*

Query by example (Qbe) refers to a kind of content-based retrieval technique where the target descriptors are extracted from examples, so users do not need to know about them. The idea of QbE has become widespread for music (e.g., singing a melody, or recording a fragment of audio to retrieve a song title and author). The same idea has been tried for general audio, where users try to imitate the sound they are looking for [6, 15] (see [62] for an example implementation using Freesound). For Qbe to work, the query example must be analyzed by the same extractor program that has been used to create the database, since small differences in parameters can lead to complete different ranges for descriptors. Again it is also possible to devise ideal targets by experts, for example, specifying a given value for pitch. Similarity queries will normally return a list of records ordered by similarity to the target. This can be seen as a nearest-neighbors search in the vector space. Common algorithms for nearest-neighbor search are KD-trees, R-trees, and ball trees. Relaxation of the problem to approximate nearest neighbors (where the returned vector is not guaranteed to be the nearest one to the target) may perform several orders of magnitude faster, and thus is indicated for large data sets and high-dimensional spaces.

The accompanying code examples include a simple implementation of both descriptor and nearest-neighbors queries. We first query the database to get general statistics about extracted audio features which allows to observe their distribution. We focus on the spectral centroid feature, which is a rough indicator of how energy is distributed across the frequency spectrum. The database is then queried for sounds with a centroid below 50 Hz, which returns a roar sound with low frequencies, and a nearest-neighbors ball tree algorithm is used to find the ten nearest neighbors using MFCC statistics. This returns a list with mostly roar sounds and also some dog barks.

10.4.5 *Audio Fingerprints and Thumbnails*

An audio fingerprint summarizes an audio recording into a small description (typically an alphanumeric string) that is ideally unique. This is used to identify copies of the same recording, since applying the same algorithm should result in the same fingerprint. Systems are often designed to be robust to some distortions, such as ambient noise or reverberation, but in general fingerprinting only works for copies of the same recording (i.e., the same waveform), as opposed to multiple recordings of similar sounds, such as a given utterance or a musical piece. The techniques used for fingerprinting are generally based on feature extraction as described above, typically with a more complex step of summarization of the time series of audio features. For example, vector quantization or hidden Markov models can be used in order to obtain a short and hopefully unique representation (see [12] for a review).

Fingerprinting can be seen as a special case of hashing. Generic hashing algorithms can be used to find and prevent exact duplicates of the same file in the database (e.g., MD5 is used in Freesound). Audio fingerprinting was developed mostly for music but it can be used also for general audio content (e.g., commercial monitoring [31]). Fingerprinting has also been used for identifying room ambiance [4], so it could be used to group recordings from the same location in an audio database. Finally, fingerprinting-like hashing has been proposed also as experimental indexing for creative applications [16].

While fingerprints can be seen as summaries that uniquely identify audio recordings for machines, audio thumbnails can be seen as sound fragments that humans can use as previews to identify and remember recordings. Such previews are required for browsing audio databases or analyzing search results. For musical audio, the most common approach is identifying frequently repeated passages[2]. Contrastingly, for environmental sound, particularly for long recordings of sound scenes, it is more useful to apply some detection strategy in order to find salient events [82].

10.5 Further Approaches to Sound Retrieval

In the previous sections we have introduced standard sound retrieval strategies based either on metadata or on audio information. In this section we introduce some advanced strategies which are not as common as those introduced in the previous sections but are also very relevant for sound retrieval.

If we have a closer look at the metadata-based strategies described in Sect. 10.3, we will see that none of them are in fact particular or restricted to the sound sharing domain. In other words, no knowledge specific to the audio domain is used for any of the scoring functions, faceting or tagcloud examples shown above. The inclusion of domain-specific knowledge is therefore something that can be considered for enhancing sound retrieval strategies [48]. A simple form of domain-specific knowledge that is relevant in sound retrieval are, for example, taxonomic classifications of sound events as seen in Chap. 7. Such taxonomies can be used at different stages of the information retrieval process. For example, a taxonomy can be used to perform domain-specific query expansion [7] and increase in this way the recall of search results. Taxonomies can be used to group search results in specific concepts and present them accordingly [35].

Another more complex form of domain-specific knowledge is that represented by ontologies [18]. Ontologies provide, for a given domain, an unambiguous formalization of its concepts, entities, and their relations. Besides the work by Nakatami and Okuno [49] in which an ontology for sound is provided, the use of ontologies has not been much explored in the field of sound scene analysis. Typically, simpler forms for representing structured domain-specific knowledge are used as exemplified by Gaver's *map* of everyday sounds [22] and the recent Urban Sound Taxonomy [65]. Nevertheless, one advantage of using ontologies is that

content can be annotated with labels which feature a very specific semantic meaning. Hence, where tagging systems feature free-form textual labels with no predefined semantic meaning, ontologies feature detailed concept hierarchies interlinked with semantically meaningful relations. The accurateness and rigidity of ontologies is often opposed to the flexibility and ambiguity of tagging systems, but these can also be complementary [40, 47]. One common approach in this direction is the mapping of user-provided tags with specific concepts of an ontology. This allows tackling typical synonymy and ambiguity problems of tagging systems, but requires methods for automatically matching tags with concepts of the ontology [1, 53]. Ontologies can also be used in a sound retrieval context for optimizing sound annotations provided by users. For example, an ontology that embeds information about types of sounds and their relevant characteristics can be used during an annotation process to suggest users to provide annotations about particularly relevant information facets [19].

In the context of online sharing platforms in which users contribute and consume audio content we can also think of retrieval strategies that take advantage of user behavior information. The most prominent example of this type of retrieval strategies are recommendation systems [56]. Recommendation systems can be defined in different ways, but in the context of sound sharing a common application is the recommendation of potentially relevant sounds for a user given previous sounds that the user has retrieved. This problem is typically approached using collaborative filtering techniques [66]. Such techniques are able to recommend items to a user based on items that other similar users interacted with in the past. For example, if user A has downloaded sounds 1, 2, and 3 and user B has downloaded sounds 1 and 3; the recommendation system could recommend sound 2 to user B. Collaborative filtering techniques can be used for discovery through sound recommendation in a way that evolves along with users' activity. The more users interact with sounds, the more information the system has to perform better informed recommendations.

With respect to content-based approaches, sound retrieval often benefits from machine learning approaches that map low-level features to more intuitive representations. Machine learning algorithms for retrieval can be generally classified between supervised and unsupervised. As shown elsewhere in this book, supervised learning approaches have applications in acoustic event classification (Chap. 5), annotation (Chaps. 6 and 7), and detection (Chap. 8) among others. Its application to sound retrieval often implies dealing with scalability both in terms of computational cost and concept generalization. For example, the statistics of a large set of features have been used along with K-NN classifiers for large-scale applications [13]. Another example approach for tackling concept generality is combining classification based on existing taxonomies with free text queries [63].

Unsupervised machine learning methods are well suited for browsing and discovery, typically by using clustering to discover underlying groupings in the database. The most common approach is to map a collection of sounds to two-dimensional space. Self-organizing maps (SOM) were used in a number of efforts for this purpose [9, 28, 51, 52]. Another approach is using graph layout algorithms

for visualizing nearest-neighbor graphs [69]. Nearest-neighbors graphs can also be clustered using graph clustering to provide unsupervised hierarchical organizations [60]. These browsing and discovery mechanisms do not require a textual query to initially filter content, but can be effectively used in combination with textual queries or supervised approaches to provide focused unsupervised interfaces. These are especially useful when many kinds of sounds are mixed in the database. Two recent examples of such interfaces are shown in Fig. 10.4. The first example, *Floop*¹⁰ (top), is an experimental system for graphically browsing rhythmic sounds. Rhythmic sounds are detected and classified in an unsupervised fashion using the Beat spectrum [21], a classic descriptor that estimates the main periodicities in any kind of sound [61]. A force-directed graph layout is used to organize a nearest-neighbors graph (computed from content-based timbral similarity) for a subset of sounds that share the same repetitive period and therefore can be played rhythmically together. The second example (Fig. 10.4, bottom) shows an interface for exploring an audio database in which the search results of a given textual query are organized according to timbral similarity. Similarly to previous work by Heise et al. [28], results are displayed in a map that can be explored and in which sounds can be listened to.¹¹ The map is computed using the t-SNE [41] dimensionality reduction technique on MFCC audio descriptors. Closer sounds in the map have closer timbral similarity. In this way search results are placed in different parts of the map and users can browse content by combining the semantic properties specified via the text search and the timbre characteristics represented in the map of results.

10.6 Conclusions

The increasing popularity of sound sharing and the growing capabilities of portable recording devices, including mobile phones, pose new challenges for sound retrieval techniques. Sound retrieval is therefore a timely topic which will probably attract more and more attention in the coming years.

In this chapter we have introduced the most important concepts related to sound sharing and retrieval and have described the different ways in which content from an audio database can be indexed, searched, and navigated. We have illustrated the different parts of a sound retrieval system with code examples showing the creation of an audio database and the addition of both metadata-based and audio-based retrieval functionalities. This code can easily be extended to incorporate more features and further experiment with sound retrieval techniques.

The introduction given in this chapter should be understood as a starting point for future developments. In particular, promising research directions such as the use of deep learning for the annotation of audio content and the use of domain-specific ontologies for structuring metadata are likely to play an important role in future sound sharing and retrieval systems.

¹⁰<https://labs.freesound.org/floop/>.

¹¹<https://ffont.github.io/freesound-explorer/>.

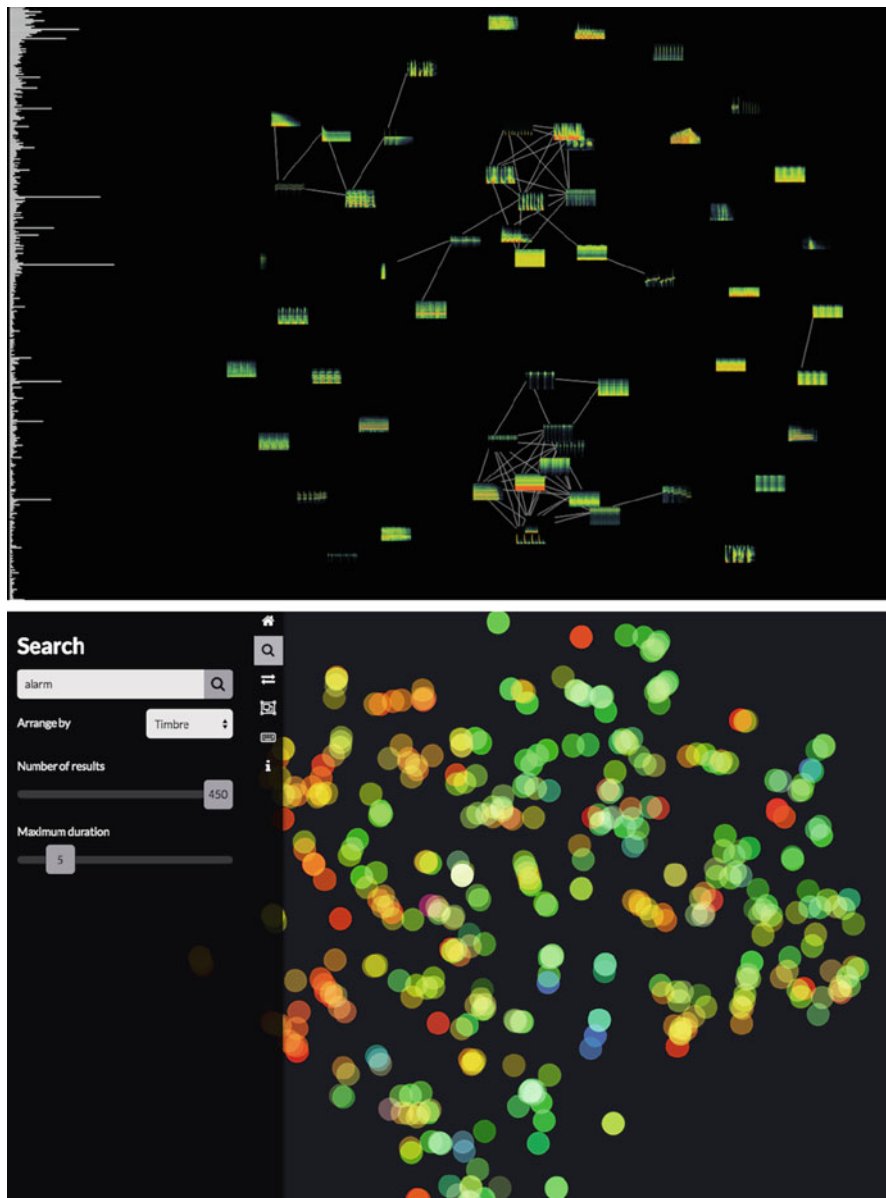


Fig. 10.4 Examples of two interfaces that exploit audio-features information to display sounds in a two-dimensional space. The *top figure* shows Floop, an interface that organizes rhythmic sounds according to an estimated periodicity. At the *left*, an interactive histogram indicates the number of sounds available for each rhythmic cycle duration. When clicking on one of the bars, the corresponding sounds are displayed and organized by timbral similarity. The *bottom figure* shows a map in which sounds are organized by timbral similarity. Users can introduce some textual query terms which are used to query Freesound and the results are displayed in a map where each circle represents a sound. Closer circles in the map tend to sound more similar than circles which are farther away

References

1. Angeletou, S., Sabou, M., Motta, E.: Semantically enriching folksonomies with FLOR. In: Proceedings of the European Semantic Web Conference (ESWC) (2008)
2. Aucouturier, J.J., Sandler, M.: Finding repeating patterns in acoustic musical signals: applications for audio thumbnailing. In: Audio Engineering Society Conference: 22nd International Conference: Virtual, Synthetic, and Entertainment Audio. Audio Engineering Society, New York (2002)
3. Aucouturier, J.J., Defreville, B., Pachet, F.: The bag-of-frames approach to audio pattern recognition: a sufficient model for urban soundscapes but not for polyphonic music. *J. Acoust. Soc. Am.* **122**(2), 881–891 (2007)
4. Azizyan, M., Constandache, I., Roy Choudhury, R.: Surroundsense: mobile phone localization via ambience fingerprinting. In: Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom), pp. 261–272. ACM, New York (2009)
5. Bischoff, K., Firan, C.S., Nejd, W., Paiu, R.: Can all tags be used for search? In: Proceedings of the ACM Conference on Information and Knowledge Management (CIKM), pp. 193–202 (2008)
6. Blancas, D.S., Janer, J.: Sound retrieval from voice imitation queries in collaborative databases. In: Proceedings of the AES Conference on Semantic Audio. Audio Engineering Society, New York (2014)
7. Bodner, R.C., Song, F.: Knowledge-based approaches to query expansion in information retrieval. In: Proceedings of the Biennial Conference of the Canadian Society for Computational Studies of Intelligence (AI), pp. 146–158. Springer, New York (1996)
8. Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J.R., Serra, X.: Essentia: an audio analysis library for music information retrieval. In: Proceedings of the International Music Information Retrieval Conference (ISMIR), pp. 493–498 (2013)
9. Brazil, E., Fernstroem, M., Tzanetakis, G., Cook, P.: Enhancing sonic browsing using audio information retrieval. In: Proceedings of the International Conference on Auditory Display (ICAD), Kyoto, pp. 132–135 (2002)
10. Brossier, P.M.: The aubio library at MIREX 2006. In: Proceedings of the Music Information Retrieval Evaluation Exchange (MIREX), p. 1 (2006)
11. Bullock, J., Conservatoire, U.: Libxtract: a lightweight library for audio feature extraction. In: Proceedings of the International Computer Music Conference (ICMC), pp. 22–28 (2007)
12. Cano, P., Batlle, E., Kalker, T., Haitsma, J.: A review of audio fingerprinting. *J. VLSI Signal Process. Syst.* **41**(3), 271–284 (2005)
13. Cano, P., Koppenberger, M., Wack, N.: An industrial-strength content-based music recommendation system. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, p. 673 (2005)
14. Carpineto, C., Romano, G.: A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.* **44**(1), 1:1–1:50 (2012)
15. Cartwright, M., Pardo, B.: Vocalsketch: Vocally imitating audio concepts. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI), pp. 43–46. ACM, New York (2015)
16. Casey, M.A.: Acoustic lexemes for organizing internet audio. *Contemp. Music Rev.* **24**(6), 489–508 (2005)
17. Comajuncosas, J.M., Barrachina, A., O’Connell, J., Gaus, E.: Nuvolet: 3d gesture-driven collaborative audio mosaicing. In: Proceedings of the New Interfaces for Musical Expression Conference (NIME), pp. 252–255 (2011)
18. Fensel, D.: *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer, New York (2001)
19. Font, F.: *Tag Recommendation using Folksonomy information for online sound sharing platforms*. Ph.D. thesis, Universitat Pompeu Fabra (2015)

20. Foote, J.: An overview of audio information retrieval. *Multimed. Syst.* **7**(1), 2–10 (1999)
21. Foote, J., Uchihashi, S.: The beat spectrum: a new approach to rhythm analysis. In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)* (2001)
22. Gaver, W.W.: What in the world do we hear?: An ecological approach to auditory event perception. *Ecol. Psychol.* **5**(1), 1–29 (1993)
23. Ghias, A., Logan, J., Chamberlin, D., Smith, B.C.: Query by humming: musical information retrieval in an audio database. In: *Proceedings of the ACM International Conference on Multimedia (MM)*, pp. 231–236. ACM, New York (1995)
24. Golder, S.A., Huberman, B.A.: Usage patterns of collaborative tagging systems. *J. Inf. Sci.* **32**(2), 198–208 (2006)
25. Guo, G., Li, S.Z.: Content-based audio classification and retrieval by support vector machines. *IEEE Trans. Neural Netw.* **14**(1), 209–215 (2003)
26. Guy, M., Tonkin, E.: Folksonomies: tidying up tags? *D-Lib Mag.* **12**(1) (2006)
27. Halpin, H., Robu, V., Shepard, H.: The dynamics and semantics of collaborative tagging. In: *Proceedings of the Semantic Authoring and Annotation Workshop (SAAW)*, pp. 1–21 (2006)
28. Heise, S., Hlatky, M., Loviscach, J.: Soundtorch: quick browsing in large audio collections. In: *Proceedings of the 125th AES Convention*. Audio Engineering Society (2008)
29. Huber, D.M., Runstein, R.E.: *Modern Recording Techniques*. Taylor & Francis, London (2013)
30. Jaitly, N., Hinton, G.: Learning a better representation of speech soundwaves using restricted boltzmann machines. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5884–5887. IEEE, New York (2011)
31. Jang, D., Jin, M., Lee, J.S., Lee, S., Lee, S., Seo, J.S., Yoo, C.D.: Automatic commercial monitoring for TV broadcasting using audio fingerprinting. In: *Proceedings of the AES Conference on Audio for Mobile and Handheld Devices*. Audio Engineering Society, New York (2006)
32. Jeffries, A.: The man behind Flickr on making the service ‘awesome again’ (2013). <http://www.theverge.com/2013/3/20/4121574/flickr-chief-markus-spiering-talks-photos-and-marissa-mayer>. Last accessed 15 Nov 2016
33. Kaser, O., Lemire, D.: Tag-cloud drawing: algorithms for cloud visualization. In: *Proceedings of the International World Wide Web Conference (WWW)* (2007)
34. Krumm, J., Davies, N., Narayanaswami, C.: User-generated content. *IEEE Pervasive Comput.* **10**–11 (2008)
35. Kummamuru, K., Lotlikar, R., Roy, S., Singal, K., Krishnapuram, R.: A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In: *Proceedings of the International Conference on World Wide Web (WWW)*, pp. 658–665. ACM, New York (2004)
36. Lartillot, O., Toivainen, P., Eerola, T.: A MATLAB toolbox for music information retrieval. In: *Proceedings of the Data analysis, Machine Learning and Applications Conference*, pp. 261–268. Springer, Berlin, Heidelberg (2008)
37. Lee, K., Ellis, D.P.W.: Audio-based semantic concept classification for consumer video. *IEEE Audio Speech Language Process.* **18**(6), 1406–1416 (2010)
38. Lee, H., Pham, P., Largman, Y., Ng, A.Y.: Unsupervised feature learning for audio classification using convolutional deep belief networks. In: *Proceedings of the Neural Information Processing Systems (NIPS)*, pp. 1096–1104 (2009)
39. Lessing, L.: *Remix: Making Art and Commerce Thrive in the Hybrid Economy*. Penguin Press, Harmondsworth (2008)
40. Limpens, F., Gandon, F.L., Buffa, M.: Linking folksonomies and ontologies for supporting knowledge sharing: a state of the art. Tech. rep., Institut National de Recherche en Informatique et Automatique (INRIA) (2009)
41. Maaten, L.V.D., Hinton, G.: Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**(Nov), 2579–2605 (2008)
42. Macgregor, G., McCulloch, E.: Collaborative tagging as a knowledge organisation and resource discovery tool. *Libr. Rev.* **55**(5), 291–300 (2006)

43. Marcell, M.M., Borella, D., Greene, M., Kerr, E., Rogers, S.: Confrontation naming of environmental sounds. *J. Clin. Exp. Neuropsychol.* **22**(6), 830–864 (2000)
44. Marlow, C., Naaman, M., Boyd, D., Davis, M.: HT06, Tagging paper, taxonomy, Flickr, academic article, to read. In: *Proceedings of the ACM Conference on Hypertext and Hypermedia (Hypertext)*, pp. 31–41 (2006)
45. Mathieu, B., Essid, S., Fillon, T., Prado, J., Richard, G.: YAAFE, an easy to use and efficient audio feature extraction software. In: *Proceedings of the International Music Information Retrieval Conference (ISMIR)* (2010)
46. McFee, B., Raffel, C., Liang, D.: librosa: Audio and music signal analysis in python. In: *Proceedings of the Python in Science Conference (SciPy)* (2015)
47. Mika, P.: Ontologies are us: a unified model of social networks and semantics. *Web Semant.: Sci. Serv. Agents World Wide Web* **5**(1), 5–15 (2007)
48. Nagypál, G.: Improving information retrieval effectiveness by using domain knowledge stored in ontologies. In: *Proceedings of the OTM Confederated International Conferences - On the Move to Meaningful Internet Systems*, pp. 780–789. Springer, New York (2005)
49. Nakatani, T., Okuno, H.G.: Sound ontology for computational auditory scene analysis. In: *Proceedings of the Innovative Applications of Artificial Intelligence Conference (IAAI)*, pp. 1004–1010 (1998)
50. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. Tech. rep., Stanford InfoLab (1999)
51. Pampalk, E., Rauber, A., Merkl, D.: Content-based organization and visualization of music archives. In: *Proceedings of the ACM International Conference on Multimedia (MM)*, pp. 570–579. ACM, New York (2002)
52. Pampalk, E., Hlavac, P., Herrera, P.: Hierarchical organization and visualization of drum sample libraries. In: *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Naples, pp. 378–383 (2004)
53. Passant, A., Laublet, P., Breslin, J.G., Decker, S.: A URI is worth a thousand tags: from tagging to linked data with MOAT. In: *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, p. 279 (2011)
54. Pedregosa, F., Varoquaux, G.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
55. Peeters, G.: A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Tech. rep., IRCAM (2004)
56. Resnick, P., Varian, H.R.: Recommender systems. *Commun. ACM* **40**(3), 56–58 (1997)
57. Robertson, S., Zaragoza, H.: The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.* **3**(4), 333–389 (2009)
58. Robu, V., Halpin, H., Shepherd, H.: Emergence of consensus and shared vocabularies in collaborative tagging systems. *ACM Trans. Web* **3**(4) (2009)
59. Rohloff, K., Dean, M., Emmons, I., Ryder, D., Sumner, J.: An evaluation of triple-store technologies for large data stores. In: *Proceedings of the OTM Confederated International Conferences - On the Move to Meaningful Internet Systems*, pp. 1105–1114. Springer, New York (2007)
60. Roma, G.: Algorithms and representations for supporting online music creation with large-scale audio databases. Ph.D. thesis, Universitat Pompeu Fabra (2015)
61. Roma, G., Serra, X.: Music performance by discovering community loops. In: *Proceedings of the Web Audio Conference (WAC)*, Paris (2015)
62. Roma, G., Serra, X.: Querying Freesound with a microphone. In: *Proceedings of the Web Audio Conference (WAC)* (2015)
63. Roma, G., Janer, J., Kersten, S., Schirosa, M., Herrera, P., Serra, X.: Ecological acoustics perspective for content-based retrieval of environmental sounds. *EURASIP J. Audio Speech Music Process.* **2010**, 1–11 (2010)
64. Salamon, J., Bello, J.P.: Feature learning with deep scattering for urban sound analysis. In: *Signal Processing Conference (EUSIPCO), 2015 23rd European*, pp. 724–728. IEEE, New York (2015)

65. Salamon, J., Jacoby, C., Bello, J.P.: A dataset and taxonomy for urban sound research. In: Proceedings of the ACM International Conference on Multimedia (MM), pp. 1041–1044 (2014)
66. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 285–295. ACM, New York (2001)
67. Schwarz, D.: Corpus-based concatenative synthesis. *IEEE Signal Process. Mag.* **24**(2), 92–104 (2007)
68. Schwarz, D., Cahen, R., Britton, S.: Principles and applications of interactive corpus-based concatenative synthesis. *J. d'Informatique Musicale* **1** (2008)
69. Schwarz, D., Schnell, N.: Sound search by content-based navigation in large databases. In: Proceedings of the Sound and Music Computing Conference (SMC), p. 1 (2009)
70. Sen, S., Lam, S., Rashid, A., Cosley, D.: Tagging, communities, vocabulary, evolution. In: Proceedings of the Conference on Community Supported Cooperative Work (CSCW), pp. 181–190 (2006)
71. Shirky, C.: Ontology is overrated: Categories, links, and tags (2005). http://www.shirky.com/writings/ontology_overrated.html. Last accessed 15 Nov 2016
72. Singhal, A.: Modern information retrieval: a brief overview. *Bull. IEEE Comput. Soc. Tech. Commun. Data Eng.* **24**(4), 35–43 (2001)
73. Smith, T.: The social media revolution. *Int. J. Mark. Res.* **51**(4), 559–561 (2009)
74. Sood, S.C., Owsley, S.H., Hammond, K.J., Birnbaum, L.: TagAssist: automatic tag suggestion for blog posts. In: Proceedings of the International Conference on Weblogs and Social Media (ICWSM), pp. 1–8 (2007)
75. The YouTube Team: Here's to eight great years (2013). <http://youtube-global.blogspot.com/2013/05/heres-to-eight-great-years.html>. Last accessed 15 Nov 2016
76. Tunkelang, D.: Faceted search. *Synth. Lect. Inf. Concepts Retr. Serv.* **1**(1), 1–80 (2009)
77. Tzanetakis, G., Cook, P.: Marsyas: a framework for audio analysis. *Organised Sound* **4**, 169–175 (2000)
78. Wagner, C., Strohmaier, M., Huberman, B.: Semantic stability and implicit consensus in social tagging streams. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 735–746 (2014)
79. Wahlforss, A.L.: SoundCloud is 5! (2013). <http://blog.soundcloud.com/2013/11/13/soundcloud-is-5/>. Last accessed 15 Nov 2016
80. Wikipedia: Remix culture (2014). https://en.wikipedia.org/wiki/Remix_culture. Last accessed 15 Nov 2016
81. Zils, A., Pachet, F.: Musical mosaicing. In: Proceedings of the International Conference on Digital Audio Effects (DAFx), p. 135 (2001)
82. Zlatintsi, A., Maragos, P., Potamianos, A., Evangelopoulos, G.: A saliency-based approach to audio event detection and summarization. In: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European, pp. 1294–1298. IEEE, New York (2012)

Chapter 11

Computational Bioacoustic Scene Analysis

Dan Stowell

Abstract The analysis of natural and animal sound makes a demonstrable contribution to important challenges in conservation, animal behaviour, and evolution. And now bioacoustics has entered its big data era. Thus automation is important, as is scalability in many cases to very large amounts of audio data and to real-time processing. This chapter will focus on the data science and the computational methods that can enable this. Computational bioacoustics has some commonalities with wider audio scene analysis, as well as with speech processing and other disciplines. However, the tasks required and the specific characteristics of bioacoustic data require new and adapted techniques. This chapter will survey the tasks and the methods of computational bioacoustics, and will place particular emphasis on existing work and future prospects which address scalable analysis. We will mostly focus on airborne sound; there has also been much work on freshwater and marine bioacoustics, and a small amount on solid-borne sounds.

Keywords Animal communication • Vocalisation • Ecoacoustics • Bioacoustics • Bird • Sound similarity • Species identification • Automatic species recognition • Natural sound • Soundscape • Acoustic monitoring • Passive acoustic monitoring • Animal calls • Vocal sequences

11.1 Introduction

Animals make use of sound for communication and exploration. Sound enables rapid transfer of information with no need for visual contact with the receiver, which is advantageous in dense forest, in nighttime activity, and over long distances, both in the air and underwater. As scientists, ecologists, and technologists, we can make use of such sound to gather information about animals for a wide variety of important tasks.

D. Stowell (✉)

Machine Listening Lab, Centre for Digital Music, Queen Mary University of London,
London E1 4NS, UK

e-mail: dan.stowell@qmul.ac.uk

Bioacoustics is a term that covers a wide multidisciplinary span of the study of sound in biological contexts, including topics such as the mechanical propagation of sounds through the environment, the sound production mechanisms of animals, and the phenomenology and neurology of sound perception in various species. Bioacoustics is increasingly significant to biodiversity [40]. Many species and ecosystems are threatened by human populations, by climate change and by natural processes [57, 84], and a diverse array of projects now makes use of automatic and semi-automatic bioacoustic analysis for monitoring [26, 54, 93]. Bioacoustic analysis is also key to the scientific understanding of issues such as animal communication, speciation and cultural evolution, and to the management of natural sound archives.

So which types of sound do we wish to analyse? The sounds that animals make are extraordinarily diverse. To give some examples: many mammals vocalise in a manner roughly similar to human vowel sounds, resulting in sounds as harmonic “stacks” with formant-like resonances; a familiar example is the howling of dogs or wolves (Fig. 11.1a). The group dynamics of animals producing these overlapped

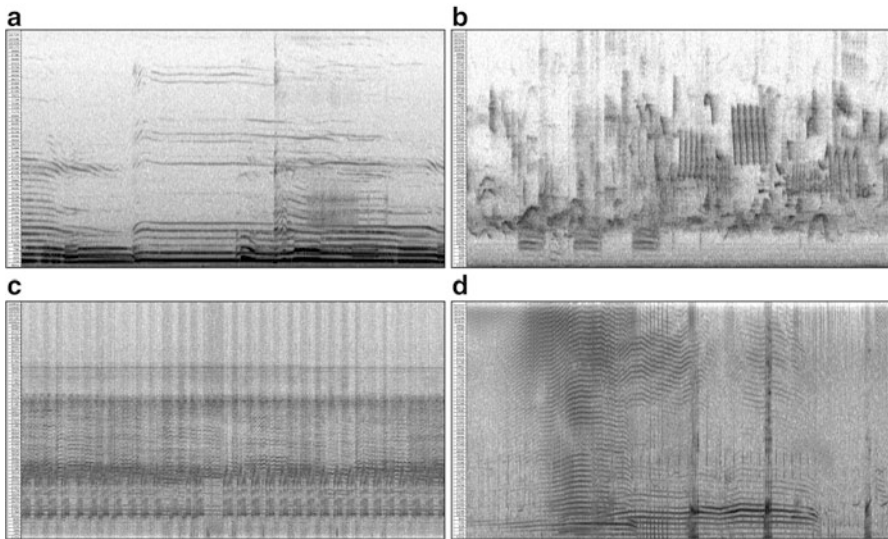


Fig. 11.1 Spectrograms illustrating sound scenes of interest in bioacoustics. Each one shows an excerpt of approx 10–20 s, 0–11 kHz. **(a)** Small pack of wolves howling together, Ähtäri, Finland. *Source: Freesound 243495, YleArkisto.* **(b)** “Dawn chorus” of many bird species, Devon, England. *Source: Freesound 275189, odilonmarcenaro.* **(c)** Frogs (lower pitch) and insects (higher), Tybee Island, Georgia. *Source: Freesound 238878, Danjocross.* **(d)** Risso’s and Pacific White-sided dolphins, off Monterey, USA. Half-speed recording. *Source: Freesound 53414, aguasonic*

sounds are an interesting challenge [34]. Some birds also produce harmonic sounds, but many also produce relatively pure tonal sounds (by using the vocal tract resonance actively to enhance the energy of the fundamental) or very noise-like sounds; songbirds also have dedicated muscles to enable them to perform complex sounds with very rapid frequency modulations (Fig. 11.1b). Other taxa such as amphibians and insects produce simpler, more stereotyped vocal units; many of these also engage in co-ordinated group calling in which it is difficult to identify each individual's sound (Fig. 11.1c). Many whales and dolphins produce variable-pitch harmonic sounds as well as clicks and buzzes (Fig. 11.1d). Bioacousticians typically have to deal with a wide array of sound types—harmonic or otherwise, percussive or extended, stereotyped or highly variable—though the picture can usually be simplified in single-species studies.

In this chapter we will focus particularly on how modern computational methods can contribute to solving problems in bioacoustics, through various forms of acoustic scene analysis. The computational aspect is important because of the increasingly large amounts of data available to bioacousticians, making automation a necessity, but also because modern computational techniques provide new tools to help us gain new insights from available data, whether that be analysing a single example in fine detail or data mining a large collection.

We start from a task-focussed perspective. This will make clear the connections with, but also the differences from, scene analysis paradigms considered elsewhere in this book. We start in Sect. 11.2 with a tour of various tasks in bioacoustics and some existing approaches to them. In Sect. 11.3 we review cross-cutting methodological issues such as measuring similarity and dealing with sound sequences. In Sect. 11.4 we focus particularly on large-scale processing, important for big data and for many bioacoustic systems to be deployed in the field. Finally in Sect. 11.5 we consider open problems and general perspectives on computational bioacoustics.

11.2 Tasks in Bioacoustics

11.2.1 *Population Monitoring, Localisation, and Ranging*

Animal population monitoring has long been recognised as an important task. Whether focussed on a particular species (for its importance or its representativeness) or on all members of an ecosystem, many decades of continued effort have been invested in obtaining good estimates of the numbers of individual animals and how they vary by time and by geographical location [84]. Much of this has been based on manual observation. The prospect of automation offers the potential for larger-scale and long-term monitoring, even in inhospitable environments [47].

A crucial issue is the level of accuracy of automatic monitoring, in general and across varying weather conditions [8, 47, 77]. Automatic methods are rarely as precise as manual monitoring by an expert. However, they do offer two relative

benefits over human observers. Firstly, observations can often be independently verified via the inspection of sensor data records. Secondly, automatic methods may be able to give a much clearer indication of their level of confidence in their decisions than can humans, and this probabilistic information can feed through usefully into analysis.

Statistical ecology is a well-developed discipline with models for estimating spatio-temporal population distributions of animals from manual observation data. These manual observation data often consist of observations (of calls or of individuals) along with bearings or distance estimates, and these are integrated using frameworks such as spatially explicit capture–recapture (SECR), distance sampling, or occupancy models [47]. These frameworks account for various factors such as missed detections and the limited area sampled. Thus, for bioacoustic monitoring, it is desirable not just to detect the presence of a sound, but to estimate qualities such as distance and bearing, and ideally to cluster vocalisations according to the individual who produced them. These attributes are not commonly handled in general-purpose audio algorithms. Indeed identifying individuals is a rather difficult task to automate in general, and so the most direct route to automation is to use methods based on call detections without individual ID.

Using acoustics to estimate the location (or the distance, the bearing) of a vocalising individual is thus useful for population monitoring. It is also useful for investigating animal behaviour (although in that case, GPS or other tracking of focal individuals may be a more direct route). The most robust approaches to acoustic estimates of location make use of multiple simultaneous recordings, from microphones arranged in a fixed array of known dimensions. This enables localisation by triangulation based on the speed of sound relative time-of-arrival of a sound at each of the microphones [4]. Various types of microphone array have been deployed and tested for localising animal sounds, on land [6, 36, 50, 81] and underwater [26], and this remains an active area of development.

In many cases a microphone array is not available but rather a single-channel recording. This may be because the data is from archives, but it will continue to be the case even for new recordings because of the advantages such as cost or equipment bulk, and data-gathering from crowdsourcing initiatives. Distance estimation from a single microphone is difficult. In general the estimate makes use of how a sound changes as it propagates through the air or water: as a sound wave travels outward from its source in a spherical radiation its energy per unit area decreases as $1/r^2$, and other atmospheric and refractive effects can modify the sound differentially at different frequencies [7, 58]. For example, making use of these propagation characteristics Dawson and Efford developed an extension of SECR useful for automatic surveys, using the signal power of a vocalisation as a proxy for its distance [17].

11.2.2 Species and Subspecies Identification

The discrimination of different animal species and subspecies is clearly needed in the majority of bioacoustic monitoring projects, for example, to avoid estimates of one species' population density being confounded by detections of some other species. The only exception is in studies of general ecosystem or acoustic richness (Sect. 11.3.5). Thus, various projects have sought to develop species identification as a classification task, using the kind of acoustic data that might be collected in a large-scale monitoring scenario [27, 54]. In developed countries the set of terrestrial fauna to be detected is typically well specified and with sufficient data to train a classifier. However, in developing countries, rich tropical ecosystems, and inaccessible locations, the set of classes to be encountered may not be fully known. Thus unsupervised analyses or *open-set* classifier designs (which account for the potential occurrence of new unseen classes) are also appropriate.

The boundaries of “species” are not always fully settled. It is not uncommon for taxonomic research to identify new subspecies or species that were previously considered an undifferentiated part of some other species, or for the reverse to happen. This occurs partly due to updated evidence, and partly due to evolutionary processes which continue to act on populations and ecosystems [46, Chapter 10]. The behaviour of species, including communicative behaviour, has long formed part of the evidence base for taxonomy [46, Chapter 12]. These decisions can have profound consequences, since the identification of a population as a unique (sub)species can directly lead to investment of resources into conservation efforts [46, Chapter 12].

Modern data collection and computation means that computational bioacoustics increasingly has a role to play in such decisions. The issue at heart is not purely that of classification, but of delineating and interrogating the boundaries between classes. This may benefit from clustering and visualisation methods, analysing the sound units as well as their sequencing.

11.2.3 “Vocabulary” Analysis, and the Study of Invariance and Change in Animal Communication Systems

Studying the “vocabulary” and “grammar” of animal communication systems is a wide research field. It can assist with tasks already mentioned (population monitoring, species delineation) and is often a fundamental task in the characterisation of a species and its behaviour. To give one example, the zebra finch *Taeniopygia guttata* is a songbird used in a wide variety of research, and its vocal repertoire was characterised by Zann through experience and through inspection of audio examples [99]. More recently, data-driven approaches using automatic classification and clustering have refined, challenged, and quantified the repertoire (for domesticated populations of zebra finch) [20, 83].

Labelling of the individual units (“syllables”) within a vocalisation is useful in studies of animal communication and can be approached as an automatic classification task [20, 67]. However, since the ground truth set of labels is rarely known, it is often better treated as a clustering task. Giving a definitive characterisation is challenging because even within a (sub)species there are often differences between separate populations and often individual differences, meaning that generalising to a whole species is not always warranted [38]. Individual differences arise particularly in species which exhibit vocal learning [46, Chapters 3 and 4]. Characteristics can change over time due to genetic or cultural evolution [39, 97]. These factors are challenges for the development of automatic methods, but also research topics in their own right, which can be aided by computational audio analysis [38].

Vocal learning in particular is a large research topic in itself, not least because the human capacity for language depends in large part on our species’ own vocal learning ability, which evolved separately from that in songbirds and some other taxa. To answer research questions about what one songbird learns from the sounds around it, it is useful to develop measures of acoustic similarity which aim to highlight physical and perceptual relationships between different sound examples (Sect. 11.3.3) [44].

The examples in Fig. 11.1 show that many animals vocalise together with others—whether these be mating partners, group members, rivals or collaborators, same or other species. To what extent do their sounds influence one another, and how might this relate to other group aspects of animal behaviour such as flocking or predator-prey dynamics? Howling wolves (Fig. 11.1a) match their timing and pitch to some extent, as do other chorusing animals. Conversely, birds in a dawn chorus (Fig. 11.1b) have been argued to avoid overlapping each other so they can maintain their respective communication channels [98]. Evidence for these inter-individual effects can be difficult to quantify. However, various approaches are now making this possible [3, 50, 59, 73].

11.2.4 Data Mining and Archive Management, Citizen Science

Archives must be able to be browsed and searched in order for their contents to be useful. The volume of natural sound data has increased exponentially in recent decades, which presents practical problems for the management of archives and research collections [63, 94]. As with other data, the full value cannot be extracted without providing tools to help users to perform data mining tasks such as searching for sound examples of a specific type or characterising the contents of sound files to determine which ones may be relevant to a query. For this, classification and clustering procedures can help to automatically annotate sound scenes and/or the events within (Chap. 8). Visualisation tools are also needed to enhance manual browsing of large audio data.

For various types of project, it is increasingly beneficial to make use of crowdsourced data (a “citizen science” approach) [94]. Compared against more controlled data collection, this raises various issues such as provenance, privacy, and varied data quality. For computational bioacoustics in particular it means that audio recordings must be analysed which were recorded under varying and often unknown conditions (hardware, background noise, distance). Tasks of particular note in citizen science include automatic labelling or validation of user-submitted labels, outlier detection, duplicate detection, and making inferences from ambiguous data.

Bioacoustics covers many more audio-based tasks than we can cover here. For example, for some species it may be possible to estimate properties of individuals such as their age, size, health, or sex, if those are reflected in characteristics such as the vocal tract shape, the frequency ranges produced, or the number and diversity of vocal units produced [9, 87, 88]. Estimating health/welfare has industrial applications in monitoring conditions for farming.

We next proceed to consider methodological aspects of relevance to people working on various bioacoustic tasks.

11.3 Methods and Methodological Issues

There are many different tasks and research questions in bioacoustics, and computational workflows vary widely. In this section we discuss various methodological issues of relevance in many areas of computational bioacoustics. We start with detection, segmentation, and classification, which are different concepts but are related and overlapping, and so we consider them together. We then consider source separation, similarity measurement, vocal sequences, holistic analysis, and visualisation, in each case focussing on the computational issues but with examples from the literature covering specific animal studies.

11.3.1 *Detection, Segmentation, and Classification*

As in other domains of sound scene analysis, many bioacoustic analyses require a sound scene to be decomposed into its component sound events, and for those events to be labelled. Procedures for detection, segmentation, and classification therefore may have much in common with procedures discussed elsewhere in this book: see in particular Chaps. 2 and 8. However, there are also important differences, which are driven by the nature of the sounds considered and by the specific questions that bioacousticians wish to ask. A particular issue we will encounter is the relative lack of ground truth, which emerges particularly when labelling the different sounds that a particular species can make.

The terms *detection* and *segmentation* are sometimes used interchangeably, both concerning the presence/absence of the sound type(s) under consideration. Although *segmentation* in general scenes can mean dividing the scene into multiple regions each of which may be of a different kind, in bioacoustics and animal communication it usually means dividing the scene into foreground (songs, calls) and background. Detection/segmentation can be conceptualised in different ways (see Fig. 8.1 in Chap. 8): some methods result in a yes/no decision about whether a sound is found in an audio clip, while some result in onset/offset regions indicating temporal location, and some result in time-frequency locations [77]. In bioacoustic analyses, many investigators annotate data by drawing time-frequency boxes on spectrogram plots. This is usually sufficient for the level of detail required and relatively intuitive and efficient for manual annotation. This method goes hand-in-hand with a commonly used method for detecting animal sounds in a sound scene: template-matching by spectrogram cross-correlation, which uses identified time-frequency patches as templates and finds matching regions in a query spectrogram [77, 85].

Although time-frequency location can be useful, for many applications the temporal location is the more important aspect of detection. It can help with navigation of long-duration audio recordings, and can also be used to divide recordings into smaller segmented regions which then go forward to further analysis such as classification. Manual segmentation has been used widely in previous decades, and is still used when a high precision is particularly important, but automatic segmentation is now common. Aside from template-matching, another common method is to select contiguous regions with relatively high energy, perhaps in a specified frequency band of interest [22, 28, 48, 69]. Ventura et al. compared various segmentation methods, and introduced a method in which temporal segmentation decisions are based on the results of *morphological filtering* (i.e. blob detection in spectrogram data) [89].

One paradigm for analysis which is rarely used for everyday sound scenes but potentially useful for bioacoustics is sinusoidal analysis or pitch tracking. Its use depends on the species to be studied: many songbirds as well as whales and dolphins produce tone-like vocalisations, although even in those taxa there are many vocalisations which do not fit a tone-like model. While acknowledging that caveat, researchers have developed various sinusoidal methods to detect and characterise vocalisations of dolphins [31, 49] and birds [14, 30]. This results in a different kind of output as in other detection methods: the objects being detected are continuous pitch tracks, or groups of these. Methods in this category might be simple or complex: recent work has found that highly simplified peak-picking methods applied to birdsong lead to surprisingly robust analysis, which is encouraging for large-scale application, though questions remain about generalisation to high-noise environments [61, 76].

Various tasks in bioacoustics are based on automatic classification applied to the regions segmented from a sound scene. One of the most widely studied is classification of species. In early research, classification was used to make a decision among a small number of potential species labels; however, in practical applications the number of species potentially present is usually large, and recent work has

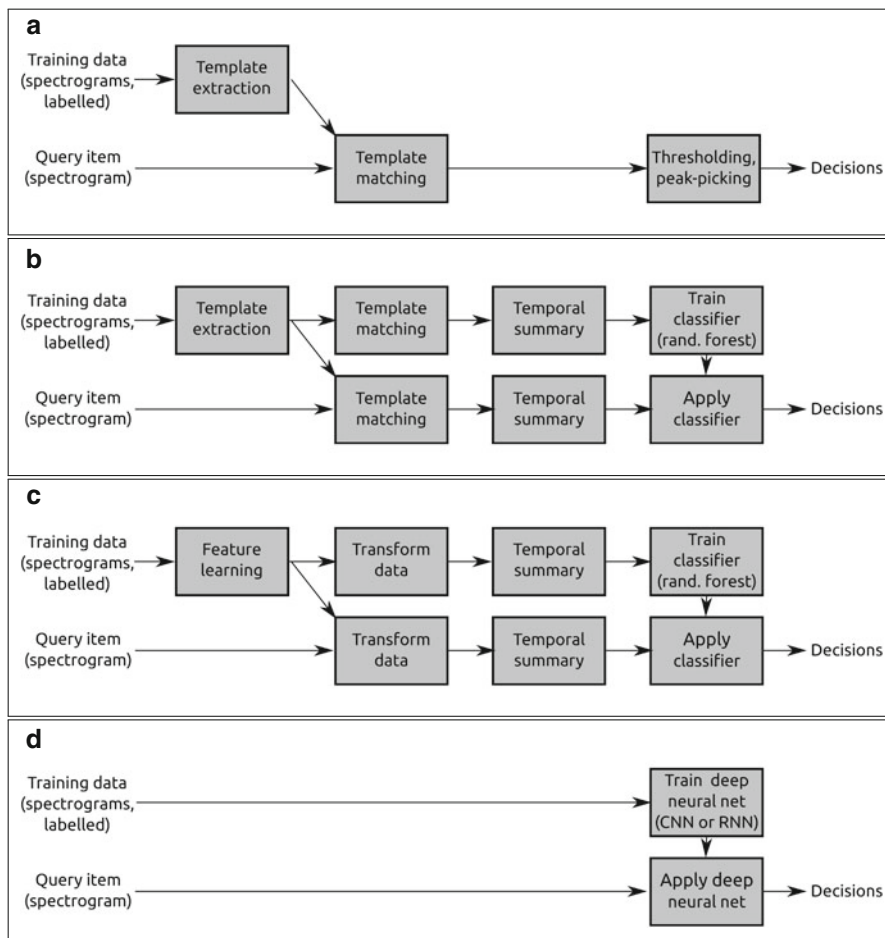


Fig. 11.2 Specific examples from current literature of workflows for automatic species classification from sound. The workflows are aligned to highlight analogies between processing steps. We have omitted preprocessing steps such as noise reduction. Examples of algorithm choices, etc., are given in brackets. **(a)** Standard template-matching, using cross-correlation or dynamic time warping [46]. **(b)** Template-matching as a feature-extraction technique [41]. **(c)** Feature learning [75]. **(d)** Deep neural network (DNN). Some DNNs operate on summary features; here we depict a DNN operating directly on spectrogram data, as in modern convolutional or recurrent neural networks [27]. Some DNNs also operate on raw waveform audio, rather than on spectrograms; at present this is unexplored for bioacoustics

been able to demonstrate strong results classifying among hundreds of species [27, 41, 76]. Figure 11.2 illustrates the processing steps involved in some examples of species classification workflows. Many of the classifiers used are similar to those used for other types of sound scene. These include support vector machines (SVMs) [10, 22], random forests [41, 76], and HMMs [30]. Up until 2016 there was very little

application of deep learning to bioacoustic classification tasks. A few researchers had used neural networks for bioacoustic analysis on a small scale [52]. This gap is notable since deep learning should be useful in many bioacoustic applications, especially large-scale ones; researchers are now starting to apply some of the deep learning methods that have already reaped benefits in speech and other audio tasks [27]. In Sect. 11.4 we will consider choice of classifier as well as other aspects from the perspective of scalability to large data volumes.

For workflows in which classification is applied to individual segmented regions of an audio scene, the overall performance of the system may be critically vulnerable to the quality of the initial segmentation procedure. For this reason, various recent methods operate on a sound recording as a whole, without attempting to remove irrelevant (e.g. silent) sound regions from consideration [10, 27, 76]. Providing that the classifier is designed/trained to allow for the irrelevant inputs, this approach can work quite generally, even in real field recordings with a variety of distractor sounds. However, there has not been a detailed study which evaluates segmentation-based versus segmentation-free modern methods for their robustness to adverse conditions such as high levels of weather noise.

Template-matching has been used in many bioacoustic projects, usually based on automatic comparison of spectrogram patches. This works well when the sounds to be identified are strongly stereotyped. It can fail to identify sounds correctly when there is a high degree of variability, such as changing duration or ordering of the units within a vocalisation (cf. Fig. 11.3). However, one recent methodological strand has repurposed template-matching for flexible large-scale classification [41]. In this approach, a library of templates is used to analyse a spectrogram. However, rather than using the strength-of-match for each template as a direct indicator of species presence, those values are interpreted as “features” to be used as input to a powerful classifier such as a random forest. If a query signal is similar but not identical to sounds from some class, then its weak matching against those sounds is a signal that a powerful classifier can use to infer an appropriate label.

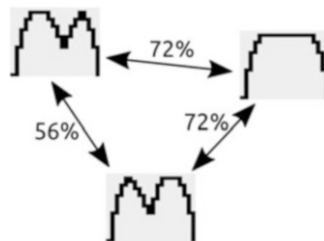


Fig. 11.3 A synthetic example to show that spectrogram cross-correlation does not always accord with perceptual similarity. We synthesised three syllables and then used spectrogram cross-correlation to compare them, normalised so that a syllable matches 100% with itself. The syllable on the *right* is the odd one out, yet it matches more strongly with the other two than they do with each other. This is because the common structure in the other two syllables takes different durations and so cannot all be aligned simultaneously

Species labelling is not the only classification task that has been studied. Labelling of individual units (“syllables”) within a vocalisation was mentioned in Sect. 11.2.3. Separately, labelling of individuals of a particular species can be useful for estimating population counts, or for analysing the interactions between individuals [62]. In practice there is an important issue to be aware of when using black-box machine-learning techniques for individual identification: sound recordings for each individual tend to have distinct background sounds as well as distinct foreground sounds, because in many cases the individuals have distinct territories or home ranges. This creates a problem of confounding that may hinder the true generalisability of the classifier, which is particularly critical if individual detection is intended to be used to identify rare circumstances such as extra-pair copulation.

One issue which has rarely been dealt with explicitly in automatic bioacoustic classification is the possibility that a sound may come from a “new” class outside the set of classes encountered during training—for example, an animal may emit an uncommon call not present in the training data; or for individual recognition we might record a previously unseen individual—and we wish these occurrences to be detected explicitly. This is sometimes called the “open set” problem, and should be a consideration for a bioacoustic recognition system deployed in the field. Ptacek et al. developed an individual classification approach designed to account for the open set problem [62]. The method makes use of a universal background model (UBM), which is a general model intended to cover any individual of the target species. During classification, the score of a query signal is evaluated against the UBM as well as the models for specific individuals, allowing the system to decide that the query comes from a known individual or from some unknown individual.

11.3.2 Source Separation

When a sound scene contains multiple animals vocalising or complex background sound, it is common to want to separate sounds out from one another so that each animal’s sounds can be analysed. Separating all sources simultaneously is a difficult task in general [90], and goes beyond the ability of animals’ own perceptual systems. In bioacoustic monitoring applications, the diverse range of foreground and background sounds encountered means that source separation on single-microphone recordings—based on models of signal properties—is of limited practical use. Instead, multi-microphone setups that allow for source separation based on spatial information (beamforming or spatial filtering) offer the most productive route for source separation in bioacoustics [3, 6, 26]. This can include focusing on specific individuals or species of interest, as opposed to analysing every single source: this is analogous to the manual technique of a recordist in the field using a parabolic microphone to record a focal individual.

Another multi-microphone approach to extracting the sound of individuals is to attach monitoring devices directly to each of the focal individuals [83]. This requires the identification and capture of the individuals in advance. This has been used widely for larger mammals such as marine mammals though for smaller animals require care with regard to what an animal can carry without impacting its behaviour or welfare [83]. Analysis of each animal's close-mic recording can reveal information not only about their own vocalisations, but also about the acoustic and behavioural context in which they occur; some work has begun to provide automatic analysis of this [72].

Alternatively to obtaining separate channels, it is prudent to consider methodologies that do not rely on the assumption that sources are not intermingled in audio. This includes methods that analyse multi-individual sequences (Sect. 11.3.4) and holistic sound scene analysis (Sect. 11.3.5).

11.3.3 Measuring Similarity Between Animal Sounds

If we are given two animal vocalisations, can we determine how similar they are? This question arises in many contexts. Often researchers wish to characterise an animal's vocal repertoire [38]. This can be aided via cluster analysis, which needs a way of measuring (dis)similarity in order to work. Other researchers wish to measure vocal learning in songbirds: how faithfully has a bird learnt to reproduce a sound made by its tutor? These questions can be asked of sequences but are also pertinent for individual vocalised units. Similarity is also used in many automatic analyses, implicitly or explicitly: some classifiers can operate directly on (dis)similarity data rather than on the items themselves.

It is important to clarify what is meant by similarity. Acoustic perception often differs from species to species. Perceptual similarity for the study species in question is often the ideal, which must be determined from studies or from psychoacoustic data [37]. Often human judgment is used as a stand-in; the extent to which this is a problem depends on the species and the application. Similarity is often judged in more pragmatic terms, for example, whether an acoustic similarity measure (such as those described below) yields a clear/useful clustering of data, or repeatable observations.

Early studies of bird sounds used easily understandable features measured directly from a vocalisation: its duration, its minimum, and maximum pitch [46]. These measures can be used to create a "space" in which similar sounds sit closely together, but in general these measures are not highly discriminative, since they do not pick up on subtleties such as the fine timing details within a call, or small modulations—subtleties which may be hard to measure directly even if they are perceptually salient.

More common is to compare the audio files against each other using a measure of similarity such as cross-correlation [46, Chapter 12]. This was originally performed using the raw waveform, but it is now very common to perform cross-correlation

of spectrograms, which has an advantage of ignoring small phase differences in the waveform (which are often irrelevant or affected by external acoustic factors), and can easily be made to focus on a frequency range of interest.

Cross-correlation is a good way to compare time series against each other when the differences are in what happens at specific times (e.g. which frequencies are heard at the *onset*). However, it is quite strict about the time basis: it does not do a good job of reflecting cases where, for example, the same sounds occur, but with different durations. In that case we might intuitively say that the sounds are rather similar, but the cross-correlation process is unable to find a way of aligning the two against each other which gives a strong match—see Fig. 11.3 for a synthetic example. Animal sounds often vary widely in exactly these duration characteristics. Hence we might seek a more flexible way of determining if two time series match up well.

One way to add flexibility is to allow the sounds to be linearly “stretched” in time relative to each other, and to find the amount of stretching that best lines up the signals, before measuring distance [82]. Dynamic time warping (DTW) is a related idea but even more general. Given two time series, it does not assume a one-to-one correspondence between time points, but seeks a flexible temporal matching which minimises an overall distance measure. The method offers flexibility in specifying the maximum allowable amount of warping, and it can be performed relatively efficiently via dynamic programming. Lachlan et al. provided evidence that it could result in similarity measurements that accorded well with the judgments of human listeners [38]. However it still often requires a larger computation than cross-correlation, as will be discussed in Sect. 11.4. This means it may become impractical for real-time or large-scale analysis, or when performing all-pairs comparison on a dataset (requiring $\frac{1}{2}(N^2 - N)$ separate comparisons where N is the number of audio clips), due to the computation time required. The benefit of DTW over simpler methods depends on the variability present in the vocalisations in question, and how much precision is required in the application. Simpler matching is often preferred for large-scale methods aimed at detecting presence/absence of bird species [41]. As an example of work at finer detail, Lachlan et al. applied DTW to both spectrogram and manual features for a close inspection of the question of whether species-universal categories exist in birdsong, finding that the evidence for universals (across different groups of the same species) is not as clear as might be assumed [38].

A different approach to similarity is based on probabilistic modelling. Instead of warping the observed features of two sounds, we can consider them as the products of some underlying generative system(s). How likely is it that they were produced by the same underlying process, or processes with similar attributes? Most commonly this is handled using a hidden Markov model (HMM) analysis [55, Chapter 10]. Each audio frame (e.g. the vector of MFCCs or spectral powers from each 10 ms chunk) of a sound signal is considered as an emission from a HMM, which can parametrically accommodate variations in duration, as well as repetitions and skipped elements. The model thus allows for even more flexibility than DTW, and is particularly useful for complex vocalisations where such flexibility is needed and sufficient training data is available. A specific HMM is trained either from one

example or a collection of similar examples, and then the similarity of a query clip is judged as the probability (the likelihood) that the model assigns to that clip. This likelihood can be reinterpreted as similarity or distance, but there is relatively little work using such distances: the most common application of this in bioacoustics is to threshold the likelihoods for detection [12, 19]. Towsey et al. [85] report in passing that they found MFCCs and HMMs to perform poorly for detection, hence their use of other methods. Ren et al. [64] argue for the HMM paradigm for general classification of animal sounds, applying it to the sounds of Asian elephants, ortolan buntings, and poultry. Wichern et al. [95] use it to construct a query retrieval system for general environmental sounds.

To summarise: when evaluating similarity by comparing the acoustic features of sounds against each other, one must remember that such acoustic similarity is only a proxy for perceptual similarity, and also that the results will be strongly affected by both the features being compared (e.g. syllable pitch/duration? spectrogram pixels?) and the similarity measure used (e.g. Euclidean distance, cross-correlation, DTW, HMM).

11.3.4 Sequences of Vocalisations

Many bioacousticians are interested in the sequencing of individual units of vocalisation—roughly equivalent to the sequencing of individual words in human speech, and with only approximate consensus as to what constitutes a “unit”—for a useful review of this topic, see [32]. The interest in call sequencing often comes from a desire to understand the communication systems being used by animals. Here we consider computational aspects and full automation of such analysis. For computational bioacoustic analysis, the questions are how can sequences usefully be modelled, how can their characteristics be estimated from data, and also how can our knowledge of sequencing help with further audio analysis. The focus has often been on the sequencing within the vocalisation of an individual animal; however, we can also consider sequencing within a pair, whether parent–offspring, breeding pair, or rivalrous pair, or for a larger set of animals such as a colony.

To analyse an audio clip, it is common to use Markov models or (equivalently) n-gram models [55, Chapter 10] [32]. Note that the sequencing here is the sequencing of individual calls, and not to the lower-level sequencing of brief audio frames as has been encountered elsewhere in this book (Chaps. 2 and 8). Thus, the audio stream must first be segmented into units, and those units must be labelled. In principle the labels could be continuous (analysed via a “state space model” rather similar to a Markov model; cf. Chap. 8 Sect. 8.4) but usually a discrete set of unit labels is used. The labels may be applied by manual inspection of the audio, potentially alongside other contextual information; but for automatic analysis it is common to use similarity-based clustering to analyse a set of un-annotated data, or classification if annotated examples are available (e.g. [20, 21, 38, 83]). This then converts an audio clip to a symbol sequence such as AAABBCBC. You should not lose sight of

the arbitrariness of this discretisation, and the limits it places on analysis. In many cases it is unclear whether the range of expression should be treated as categorical, continuous, or some mixture of the two. Even if unit categories do exist, there may be additional information (such as motivational state) encoded in the variations of expression, which are obscured by this conversion. The salient dimensions of variation are not always apparent to a human observer, nor can we guarantee they will be detected automatically, unless we have high confidence that our measures of similarity correspond well to the production and/or perceptual abilities of the species. Analysis based on symbol sequences can be useful, provided that the underlying assumptions required to convert an audio symbol sequence to a symbol sequence do not go unexamined.

Given a symbol sequence, the Markov modelling paradigm is common, and generally gives a useful characterisation of the basic sequential phenomena observed. Extensions of the Markov model have been explored both to bring the model closer to the presumed reality and to integrate other data. One is the *semi-Markov model* (SMM) or *explicit-duration Markov model* (EDMM), in which each symbol is not just emitted once, but repeated some number of times (governed by a suitably chosen probability distribution) before the transition to the next state. This semi-Markov or explicit-duration structure is considered in detail in Chap. 8 Sect. 8.3.2.2; Kershenbaum et al. argue that this structure is better-suited to many animal vocalisations than a basic Markov model [33]. (Please note that Kershenbaum et al. use unconventional terminology: they describe their explicit-duration model as a “renewal process”, but this term actually describes a slightly different model, which we consider next.) Other possible extensions include context-dependent Markov models or hierarchical Markov models, both of which make the sequence emission probabilities depend on unseen higher-level state: respectively, contextual variables or “parent” Markov models [56]. Such models might be used in cases where they match our beliefs about the particular structure of a species’ vocalisations. However, for animals which emit sequences that are complex enough to merit such analysis, there is rarely scientific consensus about what structures underlie the vocal production [1, 100].

The Markov models discussed above omit to consider one notable aspect of animal sound sequences: their timing. A transcription AAABBCBC does not tell us if the sounds occurred regularly or irregularly, fast or slow, yet it is clear from listening to animal sounds such as birdsong that there is significant structure in the timing. The timing might therefore be used as part of source separation, or identification of a species or an individual, irrespective of considerations about its meaning to the animal itself.

Autocorrelation and cross-correlation have been used to analyse the timings of vocalisations—often applied to the onset times of events, rather than to spectrograms as discussed above for detection. This produces descriptive statistics which can be used to study similarities/differences in individuals or in groups of animals [25, 59]. Alternatively, a generative model can be fit to the same type of observations, which among other things can help to clarify cause and effect [73]. Such models are of interest in animal behaviour research, in which a variety of

phenomena are studied such as duetting, social network structure, or interactions between species. For general bioacoustic sound scene analysis, these timing-based analyses also provide numbers which relate to the numbers of individuals, the density of calling, and so forth, which can be used as features taken as input for other inferences such as population estimation.

A Markov model can be augmented to include timing information. This is then referred to as a *Markov renewal process* (MRP), so named because a *renewal process* is a statistical model of the gaps between events on a continuous timeline. An MRP model can be applied to a single stream of data in exactly the same way as a Markov model, because the time-gap information can be thought of as just one added dimension of the Markovian observation. A small complexity is added in that the gap is typically characterised as continuous-valued while the unit is given a discrete-valued label, but as long as the tools can accommodate this, the application is then as broad as for Markov models: a model can be fit to data, can be used for sequence decoding, or for classification by selecting among MRP models with maximum likelihood, for example. Stowell and Plumbley illustrated an application of MRPs which goes beyond what can be done with a standard Markov model [74]. They considered sound scenes having multiple individuals of the same species, and thus modelled the observed vocalisation sequence as being the result of *multiple* overlapping MRPs. They demonstrated that in that case the MRP model can be used to segregate the observations into separate tracks per-individual, i.e. to cluster the calls, or to perform a kind of source separation (of events rather than of audio signals).

11.3.5 Holistic Soundscape Analysis: Ecoacoustics

We have seen that many bioacoustic analyses start by locating individual units of vocalisation within a recording, whether these be syllables or entire sequences/phrases. They then operate on these units. But identifying units is not error-free; and there may be important contextual information in the audio signal. What if we could extract the information we actually need, directly from a soundscape recording as a whole, *without* ever having to divide it up into objects?

Classifying or auto-tagging an entire sound scene is one example of such holistic analysis, and has been explored for various types of audio recording. Some systems perform multilabel classification of bird species at the holistic level, classifying without segmenting or otherwise subdividing the sound scene [27, 41, 75]. But for monitoring purposes we may wish to extract other kinds of information than class labels—such as a numerical indicator of the *health* or the *diversity* of an ecosystem.

In principle this is a regression problem, but an extremely difficult and ill-posed problem: it is not even clear to what extent these ecosystem properties are encoded in the audio. So as a step toward this goal, various researchers have developed ways to characterise the *acoustic diversity* of an ecosystem soundscape. This type of

approach has been referred to as *ecoacoustics*—a term created to emphasise the holistic soundscape-wide focus, in contrast to much bioacoustics [78].

Acoustic diversity is not in general a well-defined term. Researchers aim to develop acoustic indices which match well against the intuitions of an analyst, e.g. by ranking diverse/busy sound scenes higher than others, or by helping guide the user to the (sections of) recording containing diverse sounds worthy of inspection. These indices are evaluated by determining if they are good predictors of more grounded labour-intensive measures such as the number/diversity of species audible in a sound scene [24, 42, 80]. In general correlations are observed but at a moderate level, particularly when tested on real field data, meaning that this paradigm is not yet ready to stand as a direct proxy for species diversity, but a useful approach for scalable monitoring, preprocessing, and data mining [42].

Sueur et al. [80] defined two kinds of acoustic index, inspired by measurements previously applied to species count data (see also [79]). The α indices characterise a single audio clip, while the β indices characterise the difference between two audio clips. Both types were based on analysis of the spectral and/or temporal envelope of the overall sound scene, i.e. on relatively simple features that can be efficiently extracted from a large set of recordings. The goal was to combine both types to characterise the diversity in a set of audio clips. From that work, the measurement most commonly adopted by others has been the so-called D_f measure of the spectral dissimilarity between two audio clips:

$$D_f = \frac{1}{2} \sum_{f=0}^{F-1} \left| \frac{X_1[f]}{\sum X_1[\cdot]} - \frac{X_2[f]}{\sum X_2[\cdot]} \right| \quad (11.1)$$

where X_i is the power spectrum of clip i , having F frequency bins. This results in D_f ranging from 0 to 1. (Note that the subscripted f in D_f is distinct from the frequency index f used in the summation terms.) An advantage of β indices such as D_f over α indices is that, because they are based on differences rather than absolute values, their numerical values are much less dependent on the exact recording context and hardware, and so should be expected to lead to more robust comparisons across conditions.

Lellouch et al. [42] developed related indices. They used mel spectra rather than standard spectra as input, and defined a “cumulative frequency dissimilarity” index

$$D_{cf} = \frac{1}{F} \sum_{f=0}^{F-1} \left| \sum_{g=0}^f \frac{X_1[g]}{\sum X_1[\cdot]} - \sum_{g=0}^f \frac{X_2[g]}{\sum X_2[\cdot]} \right| \quad (11.2)$$

If the inner terms in Eq. (11.1) are interpreted as probability distribution functions over frequency, then the inner terms in Eq. (11.2) can be said to be their corresponding cumulative distribution functions. This cumulative index was introduced to allow more tolerance to slight frequency shifts between the two clips.

An α index that has been explored empirically is the “acoustic complexity index” (ACI), which is motivated by the idea that biotic sounds often contain much rapid variability in intensity [60]. Similarly to the “spectral flux” measurement known in music information retrieval [18], the ACI measures the amount of energy change from one spectrogram frame to the next:

$$\text{ACI}(f) = \frac{\sum_{t=1}^{T-1} |X[t, f] - X[t + 1, f]|}{\sum_{t=1}^{T-1} |X[t, f]|} \quad (11.3)$$

where $X[t, f]$ is the spectrogram value at time t and frequency f . The sums over t might be restricted to a time-window of interest rather than the whole audio duration. The ACI, thus calculated for each frequency band, can be summarised over frequency and/or over longer time spans. Farina et al. used the ACI to inspect temporal changes in the acoustic environment [23].

All the measures considered in this section are relatively simple calculations applied to spectrogram data. This renders them particularly vulnerable to “distractor” sounds such as weather noise (wind, rain) that may lead to a false impression of biotic activity. Weather impact on remote recording devices is a common issue in bioacoustic monitoring. This is true even for underwater sound where sounds caused by wind and by sea ice can form substantial components of the soundscape [51]. In many cases, time periods with unfavourable climatic conditions are simply removed from analysis [23]. This is a practical limitation, undesirable not least because it biases analysis: some weather conditions (which may correlate positively or negatively with animal vocalisation) will be systematically underrepresented. Future work should improve the robustness of this paradigm, either through noise reduction or through further development of the indices to be measured.

Buscaino et al. evaluated the ACI in an underwater acoustic environment [11]. In the marine case, they found that the ACI was robust and reflected well the biotic activity in the area. This is because the biotic sounds of interest were often short impulsive sounds, whereas weather or human noise (due to passing ships) was slowly varying—a situation which fits well with the ACI calculation. The authors noted that when animal sound is dense enough, it can lead to a relatively static spectral profile, and thus to an unexpectedly low ACI. To account for this they apply an amplitude threshold.

In the terrestrial environment, surveying bird communities, evidence is mixed as to the utility of holistic acoustic indices. Gasc et al. found moderate correlation between indices such as D_f and other indicators of community diversity, and concluded that they could provide an acceptable surrogate [24]. Lellouch et al., however, concluded that although such indices were useful for scalable monitoring, they were not yet ready to stand as a proxy for species diversity [42].

11.3.6 Visualisation and Data Mining

Much of the focus in this book is on fully automatic methods for sound scene analysis. However, there is also much to be gained from using computation to improve manual or semi-automatic processes, such as data mining a large sound collection for sound clips relevant to a particular research question.

A clear example is the spectrogram (also known as sonogram, when applied to sound), which is a computational technique which greatly improved manual analysis of sounds when it became widely available (see, e.g., [70, 96]). Many bioacousticians routinely scan and annotate spectrograms visually. Additional information for the user interaction can be added from procedures such as call detection. Although call detection can in principle be automatic, an interaction step is typically needed to refine the output and correct errors. These workflows are implemented in widely available software such as Raven or SongScope [19]. Interactive analysis is not merely for correcting errors, though, and can be a useful way to explore audio data while developing research questions or gaining an understanding of an acoustic environment.

There are many bioacoustic projects which record large amounts of audio, because they use multiple sensors and because they record for very long durations (e.g. years). It is difficult to navigate such long-duration audio using conventional visualisation such as a spectrogram, because at a low zoom level fine details such as calls can become invisible. Hence it is useful to design tools and visualisations specifically for long-duration or large amounts of audio. Towsey et al. developed an approach to long-duration false-colour spectrograms [86]. These are time-frequency plots where each pixel, instead of representing simply the energy at a particular time and frequency, represents an *acoustic index* measured at that time and frequency (Fig. 11.4). The acoustic indices used include the ACI and the D_f discussed above. A pixel might, therefore, show up brightly if there is a lot of energy variation for the time period it represents (which could be an hour, a day, or something else). Note that in this application, the acoustic indices are calculated separately for each frequency band, rather than overall. This means the output is a two-dimensional image, passing a lot of information to the user, and having a frequency axis that is easy to interpret for most people working with audio.

11.4 Large-Scale Analysis Techniques

Bioacoustics has entered its big data era. Many projects now capture many hours, days, months of audio, from multiple recording locations [2, 68, 92]. Analysis of recordings may occur off-line, but there may also be a need for real-time processing in order to make low-latency decisions, such as decisions about which audio recordings to preserve. The task is further constrained by the fact that many remote

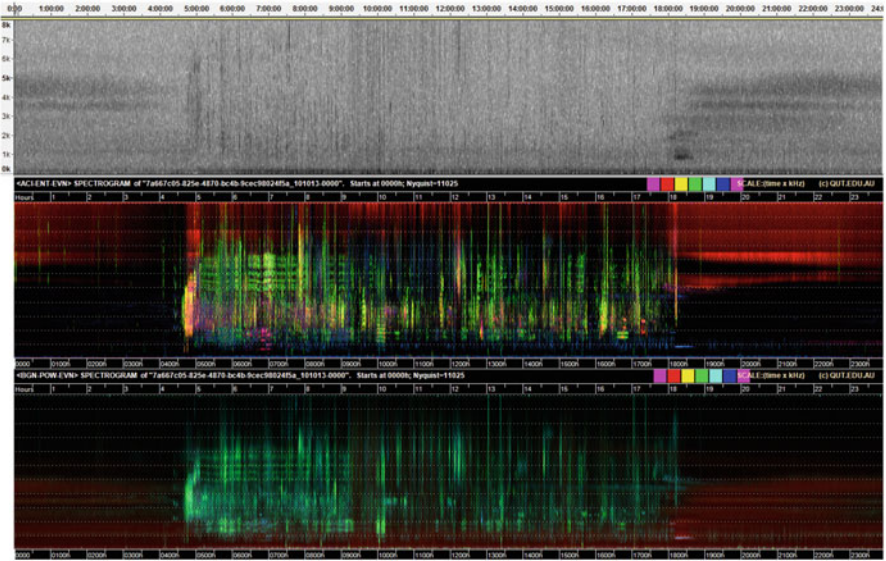


Fig. 11.4 Three spectrograms of the same 24-h recording made in bushland 30 km west of Brisbane, Australia. The recording starts and ends at midnight, with midday in the centre of each image. The time resolution is 60 s/pixel. The frequency range for the *top image* is 0–8 kHz and for the other two, 0–11 kHz. The *top grey-scale* spectrogram illustrates the “compression by averaging” performed by the audio-processing software, Audacity. The effect is to highlight only general background noise, such as the cicada chorus at 1820 h and the insect chorus tracks at night. The middle false-colour spectrogram is obtained by assigning three different acoustic indices (ACI, ENT, and EVN) to the *red*, *green*, and *blue* colour channels, respectively. The morning chorus is obvious but more surprisingly, several bird species can also be identified because their brief calls nevertheless leave similar traces in consecutive minutes of spectrogram. The *bottom false-colour* spectrogram is obtained by assigning the acoustic indices BGN, POW, and EVN to RGB, respectively. Different indices provide different “views” into the soundscape. However, in case of the lower spectrogram, two of the indices, POW and EVN are somewhat correlated and therefore less information is revealed in the false-colour rendering

monitoring units need to have low power consumption so they can be left unattended for long durations or powered, e.g. by solar panels.

Thus we will now consider a range of methods from the perspective of their scalability and real-time suitability. One way to consider scalability is via the “complexity” of an algorithm in terms of the amount of computation and/or storage it requires. In computer science, this is summarised in “big O” notation: to say an algorithm has time complexity of $O(N^2)$ (where N might, for example, be the number of datapoints) means that as N grows large, the computation time required tends to follow some constant multiplied by N^2 . For example, if we have N audio recordings and we wish to find all of the similarities between pairs of recordings, there are $\frac{1}{2}(N^2 - N)$ pairings to be considered and thus the time complexity must be at least $O(N^2)$. If an algorithm has time complexity $O(N^2)$ or higher, then it may well not be feasible to apply it to large datasets. Algorithms of lower order, such

as $O(N \log N)$ or $O(N)$ are often sought. We will use this approach in some of our discussion. For trained machine-learning methods we will distinguish between the complexity of training a system and the complexity of applying the trained system—usually the latter is the main concern. However, we also note that *asymptotic complexity* is not always a perfect guide to practical feasibility. If an algorithm requires kN bytes of memory for its calculations, then its usefulness also depends on whether the constant k is small or large.

11.4.1 *Classifiers and Detectors*

Many approaches to classification/detection are *instance-based*, meaning that in order to make a decision about a query datum, the algorithm compares it against a set of data items previously stored in memory. This is the case for simple template-matching methods which store templates of the target sound; other instance-based methods include k nearest neighbours and support vector machines (SVM). For template-matching the cross-correlation itself can be implemented through “fast” algorithms with complexity $O(TK)$ where T is the duration of the audio being searched and K the number of templates to match [43]. The memory required is $O(K)$, assuming that the audio templates have some fixed maximum duration. It can run in real time, though it is often one of the heavier computations running in a system, especially if K is large. Thus it is most appealing for applications detecting a small number of stereotyped animal sounds in which a small number of templates will suffice. For larger problems, the reference data can be pruned, using a small number of exemplars rather than all known instances. Efficient large-scale matching can also be achieved by approximate matching methods such as locality-sensitive hashing (LSH) [29]. This has been used for similarity-based audio retrieval in music informatics [13].

Comparing two audio clips via dynamic time warping (DTW) has complexity $O(T_1 T_2)$ where the T_i are the audio durations. Thus if the duration of exemplar templates has some fixed maximum duration (usually very much shorter than the audio being searched), the formal complexity of DTW is linear in the duration of the audio to be searched—the same as for cross-correlation. However, in practice it still requires notably heavier computation than cross-correlation, which may explain why it is not commonly used for large-scale analyses. Other methods with conceptual advantages over simple template-matching include hidden Markov models (HMM), non-negative matrix factorisation (NMF), and sparse representations (e.g. [66]; see also Chap. 8 for discussion of these for sound event detection). As with DTW, thus far these are little-used in bioacoustic monitoring, which may be due to unfamiliarity, but from experience it appears unlikely that they would yield large gains over cross-correlation for large-scale matching, and so the cost-benefit ratio seems to argue in favour of cross-correlation as a default choice.

Scalability is a particularly interesting question with regard to neural networks and deep learning, which is now beginning to have its impact on bioacoustics [27]. The power of modern deep neural networks (DNNs) is partly due to their use of very large datasets for training, which of course means that time taken to train a system can be very long, even if accelerated by hardware technologies such as GPU processing. This does not affect the computational complexity of applying a trained DNN: applying a DNN typically consists of a fixed network of simple calculations (such as multiplications and additions). However, a trained DNN may have many layers and a large number of parameters, which can still mean that classifying using a DNN takes a relatively large amount of computation.

Deep learning can exhibit another scalability issue for bioacoustics: since DNNs often do not show powerful performance when trained on small datasets, they may well be unsuitable for detecting/classifying sounds for which it is not feasible to collect a large amount of audio examples.

Neural nets are usually trained in streaming fashion, by exposing them iteratively to multiple small batches of examples. This allows them to train on large datasets even on systems with small memory, and also means that they are updateable: they can be trained further at any time by exposing them to further labelled data.

For further insight into the use of deep learning, the reader is encouraged to refer to Chap. 5 but also to recent work in fields such as speech recognition and to the useful handbooks on deep learning published since the advent of deep learning [53].

11.4.2 Reducing Computation Via Low-Complexity Front-Ends

If a relatively heavy-duty computation is required for good performance, or if storage is limited, then one way of improving scalability is to reduce the number of times the computation is invoked. This has often been used in bioacoustic surveys, for example, by recording on a fixed schedule such as 5 min out of every hour, which reduces the storage/analysis required by a fixed factor of 12; alternatively, energy detection or the like can be used [68]. In survey designs these decisions are made together with decisions about how often a monitoring station can be visited or can transmit its results, or how much its power supply can provide. Even if data is to be collected for an unknown amount of time, a fixed-size random sample of audio segments can be obtained through simple “reservoir sampling” algorithms [91].

In a detection-then-classification workflow, the accuracy of the detector has a strong impact on the number of times the classifier is invoked. The detector front-end can be a simple energy-based method, or something more complex [85]. As investigated by Ross and Allen, it may be desirable to use a simple low-complexity detector as the first processing stage, set to reject silences but otherwise to have a high *recall* factor, and then to refine the detection decisions using a more involved algorithm such as a random forest [65].

11.4.3 Features

Thus far we have given little consideration to the scalability implications of the choice of features for representing the audio data. This is because, in particular for template-matching approaches, the basic representation is relatively fixed, as some variant of a standard spectrogram. The audio used to produce this spectrogram may have been filtered or otherwise preprocessed; however, we note the advice of Stoddard and Owren: “In bioacoustics applications, the best advice about filtering is to use it sparingly” [71].

It is common to discard some frequency bands of the spectrogram to focus on a frequency range of interest. This can reduce the computation required, especially in single-species studies. Alternatively, to process a large number of frequency bands relatively efficiently, Ruiz et al. apply a *random projection*, a simple mathematical transformation to project the frequency bands into a lower-dimensional representation [66]. This offers the same kind of dimension-reduction benefit as does principal components analysis (PCA), but without needing to pre-analyse the dataset.

Various methods do not work directly on spectrogram features but require some processing of data into higher-level features, which might begin to have some semantic interpretation. A good example is the estimation of time-varying frequencies, known as *F₀ tracking*, *pitch tracking*, or *sinusoidal modeling*. Many such methods require a large amount of computation. Stowell et al. compared four different feature types used to extract frequency and frequency-modulation information from birdsong [76]. They found that some methods were extremely time consuming, while by contrast the simplest method performed satisfactorily as well as efficiently: it was based on picking the frequencies having peak energy in each frame of the spectrogram. For detailed analysis, peak-picking might not produce the most accurate frequency tracks, since peaks do not always correspond to the fundamental frequency; nevertheless the information recovered can be useful for large-scale as opposed to high-resolution analysis. Podos et al. made a similar observation about features based on peak-picking [61].

Many of the acoustic complexity indices (ACIs) considered in Sect. 11.3.5 are relatively simple calculations that could be performed in real time by a system. This is why they are a suitable substrate for long-duration spectrograms (Fig. 11.4). They could also be used as features for scalable machine-learning characterisation of a sound scene.

For automatic detection/segmentation of audio in a remote monitoring unit, Colonna et al. paid particular attention to the calculation of features with very low memory and computation requirements [16]. They proposed to calculate energy levels and zero-crossing rates (ZCRs) using an “exponential forgetting” method that incrementally updates the previously remembered feature value with new data.

The use of *feature learning* deserves particular scrutiny in the present discussion, with consideration of how it compares against basic features and how it relates to neural network and other algorithms. As with deep learning, feature learning

usually benefits from having a very large amount of training data available, which has an impact on the training time required but does not make a notable difference to runtime when deploying a trained system. Unsupervised feature learning via spherical k -means was introduced as a highly efficient technique to learn features from large datasets [15]. It is thus useful for analysis of very large *training* sets, finding a transformation of the data that is fitted to characteristics of the data (unlike random projections, mentioned above) and has been shown to give a strong improvement to bioacoustic signal classification [75]. A related feature learning approach was applied by Kohlsdorf et al. for dolphin vocalisations, applying (standard non-spherical) k -means feature learning to small patches of spectrogram [35]. Dictionary learning is closely related to feature learning, and has been investigated for bioacoustic monitoring [66].

The transformation that is learnt by feature learning is similar to the transformation performed by one layer of a neural network. One advantage of deep learning is that multiple layers of transformation are stacked together, progressively transforming the data to extract details. Mallat et al. introduced a different paradigm for feature extraction, the scattering transform, which is not learnt but which like a DNN consists of multiple stacked layers of non-linear transformations [45]. The scattering features have mathematical properties which are argued to capture invariances that are relevant to natural sound analysis, and have thus been explored for audio analysis. Unlike DNNs or feature learning, scattering features are not learned and so can be applied even to small datasets. Scattering features have been used as the basis for large-scale bird classification [5].

Further work will elucidate which of these approaches to feature extraction is most appropriate for the bioacoustic context, which in many cases has large data volumes but with rare or unknown sound event types which are of high importance to detect.

11.5 Perspectives and Open Problems

Through this chapter we have seen that computational bioacoustics spans a range of well-specified analysis tasks which have been studied in various ways over recent decades, and which can be improved and made efficient through modern computational techniques. There has been much work on classification and detection, for example, with connections to other work on computational audio scene analysis but having specific adaptations to the characteristics of the sounds under consideration. In this chapter we have seen tasks which computational bioacoustics has already advanced, such as ecosystem monitoring, analysis of animal vocabularies, or data mining bioacoustic archives. There are other tasks which would benefit from further development—such as making sense of sound units and their sequencing, given unlabelled audio data, and triangulating this against other data such as observed behaviour or physiology. Many such tasks are difficult to provide ground truth for, and so unsupervised analysis merits further development. There are also many

examples of projects that are not yet as automated as we might wish, because of accuracy or robustness issues. The field offers a wealth of interesting problems in which computational work on signal processing and machine learning has the potential to make great advances.

For remote monitoring, specific open problems encountered in practice include the weather robustness of acoustic detection and classification, and its generalisation to new environments without manual re-tuning [77]. Reliable estimation of location/distance, especially in mono-mic or ad-hoc mic setups, would increase the types of survey design that could be conducted [6].

Identifying individual animals from individual vocalisations remains a difficult task in general, and one that is of interest to practitioners working on different topics. The field would benefit from general approaches which can avoid confounds such as background sounds associated with territory.

The ecoacoustic question remains an open one: to what extent can acoustic measures of a soundscape be used to estimate the health or diversity of an ecosystem? So far, relatively simple measurements have been investigated, and this is appropriate since the aim is to develop generic and low-cost methods. Further improvements, without going so far as to require a full automatic transcription of every item in the soundscape, could make use of source separation, unsupervised clustering of acoustic elements, or black-box deep learning.

As discussed in Sect. 11.4, scalable methods are increasingly crucial to bioacoustic work, as data volumes grow. Deep learning has appealing characteristics in this regard, a paradigm developed in and benefitting from big-data scenarios. Deep learning has been applied in bioacoustics [27] but for high scalability and real-time processing, even more efficient feature-extraction algorithms may prove useful. Examples of alternative scalable analyses which have been applied for feature extraction in bird classification include the scattering transform which requires no training but has similarities with an unsupervised CNN analysis [5]; or spherical k-means feature learning, based on a very simple streamable algorithm which can be thought of as a simple single layer unsupervised CNN training [75].

Bioacoustics at grand scale should not obscure the continued importance of small-scale analysis. In many situations there is a need for fine-detail analysis of a single case study. In other situations there may only be a tiny amount of data available, e.g. with rare or cryptic animal species, or infrequent behaviours.

There is also need to develop methods further for model-based analysis of multi-animal interactions (within and between species, e.g. [73]). Much zoological knowledge about interactive behaviour is qualitative, or at least not yet amenable to encoding in a computational model of behaviour. With the development of such models there is potential for a beneficial feedback loop, as fitting the models to data and improving them enable us to apply these models to make inferences about bioacoustic sound scene data, such as inferring the social networks revealed in the patterns of a dawn chorus.

References

1. Abe, K., Watanabe, D.: Songbirds possess the spontaneous ability to discriminate syntactic rules. *Nat. Neurosci.* **14**, 1067–1074 (2011). doi:10.1038/nn.2869
2. Aide, T.M., Corrada-Bravo, C., Campos-Cerqueira, M., Milan, C., Vega, G., Alvarez, R.: Real-time bioacoustics monitoring and automated species identification. *PeerJ* **1**, e103 (2013). doi:10.7717/peerj.103
3. Aihara, I., Mizumoto, T., Awano, H., Okuno, H.G.: Call alternation between specific pairs of male frogs revealed by a sound-imaging method in their natural habitat. In: *Interspeech 2016*. International Speech Communication Association (2016). doi:10.21437/interspeech.2016-336
4. Anguera, X., Wooters, C., Hernando, J.: Acoustic beamforming for speaker diarization of meetings. *IEEE Trans. Audio Speech Lang. Process.* **15**(7), 2011–2022 (2007). doi:10.1109/TASL.2007.902460
5. Balestrieri, R., et al.: Scattering decomposition for massive signal classification: from theory to fast algorithm and implementation with validation on international bioacoustic benchmark. In: *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp. 753–761. IEEE, New York (2015)
6. Blumstein, D.T., Mennill, D.J., Clemins, P., Girod, L., Yao, K., Patricelli, G., Deppe, J.L., Krakauer, A.H., Clark, C., Cortopassi, K.A., Hanser, S.F., McCowan, B., Ali, A.M., Kirschel, A.N.G.: Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus. *J. Appl. Ecol.* **48**(3), 758–767 (2011). doi:10.1111/j.1365-2664.2011.01993.x. <http://dx.doi.org/10.1111/j.1365-2664.2011.01993.x>
7. Boersma, H.F.: Characterization of the natural ambient sound environment: Measurements in open agricultural grassland. *J. Acoust. Soc. Am.* **101**, 2104 (1997). doi:10.1121/1.418141
8. Borcker, A.L., McKown, M.W., Ackerman, J.T., Eagles-Smith, C.A., Tershy, B.R., Croll, D.A.: Vocal activity as a low cost and scalable index of seabird colony size. *Conserv. Biol.* (2014). doi:10.1111/cobi.12264
9. Briefer, E., McElligott, A.G.: Indicators of age, body size and sex in goat kid calls revealed using the source-filter theory. *Appl. Anim. Behav. Sci.* **133**, 175–185 (2011). doi:10.1016/j.applanim.2011.05.012
10. Briggs, F., Raich, R., Fern, X.Z.: Audio classification of bird species: a statistical manifold approach. In: *Proceedings of the Ninth IEEE International Conference on Data Mining*, pp. 51–60 (2009). doi:10.1109/ICDM.2009.65
11. Buscaino, G., Ceraulo, M., Pieretti, N., Corrias, V., Farina, A., Filicetto, F., Maccarrone, V., Grammauta, R., Caruso, F., Giuseppe, A., et al.: Temporal patterns in the soundscape of the shallow waters of a Mediterranean marine protected area. *Sci. Rep.* **6** (2016). doi:10.1038/srep34230
12. Buxton, R.T., Jones, I.L.: Measuring nocturnal seabird activity and status using acoustic recording devices: applications for island restoration. *J. Field Ornithol.* **83**(1), 47–60 (2012). doi:10.1111/j.1557-9263.2011.00355.x
13. Casey, M.A., Slaney, M.: Song intersection by approximate nearest neighbor search. In: *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, vol. 6, pp. 144–149 (2006)
14. Chen, Z., Maher, R.C.: Semi-automatic classification of bird vocalizations using spectral peak tracks. *J. Acoust. Soc. Am.* **120**(5), 2974–2984 (2006). doi:10.1121/1.2345831
15. Coates, A., Ng, A.Y.: Learning feature representations with k-means. In: *Montavon, G., Orr, G.B., Muller, K.R. (eds.) Neural Networks: Tricks of the Trade*, pp. 561–580. Springer, New York (2012). doi:10.1007/978-3-642-35289-8_30
16. Colonna, J.G., Cristo, M., Salvatierra, M., Nakamura, E.F.: An incremental technique for real-time bioacoustic signal segmentation. *Expert Syst. Appl.* (2015). doi:10.1016/j.eswa.2015.05.030

17. Dawson, D.K., Efford, M.G.: Bird population density estimated from acoustic signals. *J. Appl. Ecol.* **46**(6), 1201–1209 (2009). doi:10.1111/j.1365-2664.2009.01731.x
18. Dixon, S.: Onset detection revisited. In: Proceedings of the International Conference on Digital Audio Effects (DAFx-06), Montreal, Quebec, pp. 133–137 (2006)
19. Duan, S., Zhang, J., Roe, P., Wimmer, J., Dong, X., Truskinger, A., Towsey, M.: Timed probabilistic automaton: a bridge between Raven and Song Scope for automatic species recognition. In: Proceedings of the Twenty-Fifth Innovative Applications of Artificial Intelligence Conference, pp. 1519–1524. AAAI, Palo Alto (2013)
20. Elie, J.E., Theunissen, F.E.: The vocal repertoire of the domesticated zebra finch: a data-driven approach to decipher the information-bearing acoustic features of communication signals. *Anim. Cogn.* 1–31 (2015). doi:10.1007/s10071-015-0933-6
21. Engesser, S., Crane, J.M., Savage, J.L., Russell, A.F., Townsend, S.W.: Experimental evidence for phonemic contrasts in a nonhuman vocal system. *PLoS Biol.* **13**(6), e1002171 (2015). doi:10.1371/journal.pbio.1002171
22. Fagerlund, S.: Bird species recognition using support vector machines. *EURASIP J. Appl. Signal Process.* 38637 (2007). doi:10.1155/2007/38637
23. Farina, A., Pieretti, N., Piccioli, L.: The soundscape methodology for long-term bird monitoring: a Mediterranean Europe case-study. *Ecol. Inform.* **6**(6), 354–363 (2011). doi:10.1016/j.ecoinf.2011.07.004
24. Gasc, A., Sueur, J., Jiguet, F., Devictor, V., Grandcolas, P., Burrow, C., Depraetere, M., Pavoine, S.: Assessing biodiversity with sound: do acoustic diversity indices reflect phylogenetic and functional diversities of bird communities? *Ecol. Indic.* **25**, 279–287 (2013). doi:10.1016/j.ecolind.2012.10.009
25. Gill, L.F., Goymann, W., Ter Maat, A., Gahr, M.: Patterns of call communication between group-housed zebra finches change during the breeding cycle. *eLife* **4** (2015). doi:10.7554/eLife.07770
26. Gillespie, D., Mellinger, D.K., Gordon, J., McLaren, D., Redmond, P., McHugh, R., Trinder, P., Deng, X.Y., Thode, A.: PAMGUARD: semiautomated, open source software for real-time acoustic detection and localization of cetaceans. *J. Acoust. Soc. Am.* **125**(4), 2547–2547 (2009). doi:10.1121/1.4808713. <http://dx.doi.org/10.1121/1.4808713>
27. Goëau, H., Glotin, H., Vellinga, W.P., Planqué, R., Joly, A.: LifeCLEF bird identification task 2016: the arrival of deep learning. In: Working Notes of CLEF 2016–Conference and Labs of the Evaluation forum, Évora, Portugal, 5–8 September, 2016, pp. 440–449 (2016)
28. Härma, A., Somervuo, P.: Classification of the harmonic structure in bird vocalization. In: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), vol. 5, pp. 701–704 (2004). doi:10.1109/ICASSP.2004.1327207
29. Indyk, P., Motwani, R.: Approximate nearest neighbors: towards removing the curse of dimensionality. In: Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp. 604–613. ACM, New York (1998)
30. Jancovic, P., Kokuer, M.: Acoustic recognition of multiple bird species based on penalised maximum likelihood. *IEEE Signal Process. Lett.* 1–1 (2015). doi:10.1109/lsp.2015.2409173. <http://dx.doi.org/10.1109/LSP.2015.2409173>
31. Johansson, A.T., White, P.R.: An adaptive filter-based method for robust, automatic detection and frequency estimation of whistles. *J. Acoust. Soc. Am.* **130**(2), 893–903 (2011). doi:10.1121/1.3609117
32. Kershenbaum, A., Blumstein, D.T., Roch, M.A., Akçay, Ç.A., Backus, G., Bee, M.A., Bohn, K., Cao, Y., Carter, G., Căsar, C., et al.: Acoustic sequences in non-human animals: a tutorial review and prospectus. *Biol. Rev.* (2014). doi:10.1111/brv.12160
33. Kershenbaum, A., Bowles, A.E., Freeberg, T.M., Jin, D.Z., Lameira, A.R., Bohn, K.: Animal vocal sequences: not the Markov chains we thought they were. *Proc. R. Soc. B: Biol. Sci.* **281**(1792) (2014) 20141370. doi:10.1098/rspb.2014.1370
34. Kershenbaum, A., Root-Gutteridge, H., Habib, B., Koler-Matznick, J., Mitchell, B., Palacios, V., Waller, S.: Disentangling canid howls across multiple species and subspecies: structure in a complex communication channel. *Behav. Process.* **124**, 149–157 (2016). doi:10.1016/j.beproc.2016.01.006

35. Kohlsdorf, D., Herzog, D., Starner, T.: Feature learning and automatic segmentation for dolphin communication analysis. In: Interspeech 2016. International Speech Communication Association (2016). doi:10.21437/interspeech.2016-748. <http://dx.doi.org/10.21437/Interspeech.2016-748>
36. Kojima, R., Sugiyama, O., Suzuki, R., Nakadai, K., Taylor, C.E.: Semi-automatic bird song analysis by spatial-cue-based integration of sound source detection, localization, separation, and identification. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1287–1292. IEEE, New York (2016). doi:10.1109/IROS.2016.7759213
37. Lachlan, R.F., Nowicki, S.: Context-dependent categorical perception in a songbird. *Proc. Natl. Acad. Sci.* **112**(6), 1892–1897 (2015). doi:10.1121/1.4933900
38. Lachlan, R., Verhagen, L., Peters, S., ten Cate, C.: Are there species-universal categories in bird song phonology and syntax? A comparative study of chaffinches (*Fringilla coelebs*), zebra finches (*Taenopygia guttata*), and swamp sparrows (*Melospiza georgiana*). *J. Comp. Psychol.* **124**(1), 92 (2010). doi:10.1037/a0016996
39. Lachlan, R.F., Verzijden, M.N., Bernard, C.S., Jonker, P.P., Koese, B., Jaarsma, S., Spoor, W., Slater, P.J., ten Cate, C.: The progressive loss of syntactical structure in bird song along an island colonization chain. *Curr. Biol.* **23**(19), 1896–1901 (2013). doi:10.1016/j.cub.2013.07.057
40. Laiolo, P.: The emerging significance of bioacoustics in animal species conservation. *Biol. Conserv.* **143**(7), 1635–1645 (2010). doi:10.1016/j.biocon.2010.03.025
41. Lasseck, M.: Bird song classification in field recordings: winning solution for NIPS4B 2013 competition. In: Glotin, H., LeCun, Y., Artières, T., Mallat, S., Tchernichovski, O., Halkias, X. (eds.) *Neural Information Processing Scaled for Bioacoustics, from Neurons to Big Data, USA*, pp. 176–181 (2013). http://sabiiod.org/NIPS4B2013_book.pdf
42. Lellouch, L., Pavoine, S., Jiguet, F., Glotin, H., Sueur, J.: Monitoring temporal change of bird communities with dissimilarity acoustic indices. *Methods Ecol. Evol.* **5**(6), 495–505 (2014). doi:10.1111/2041-210X.12178
43. Lewis, J.: Fast normalized cross-correlation. *Vis. Interface* **10**(1), 120–123 (1995)
44. Lipkind, D., Tchernichovski, O.: Quantification of developmental birdsong learning from the subsyllabic scale to cultural evolution. *Proc. Natl. Acad. Sci.* (2011). doi:10.1073/pnas.1012941108
45. Mallat, S.: Group invariant scattering. *Commun. Pure Appl. Math.* **65**(10), 1331–1398 (2012). <http://arxiv.org/abs/1101.2286>
46. Marler, P.R., Slabbekoorn, H.: *Nature's Music: The Science of Birdsong*. Academic Press, New York, MA (2004)
47. Marques, T.A., Thomas, L., Martin, S.W., Mellinger, D.K., Ward, J.A., Moretti, D.J., Harris, D., Tyack, P.L.: Estimating animal population density using passive acoustics. *Biol. Rev.* (2012). doi:10.1111/brv.12001
48. McIlraith, A.L., Card, H.C.: Birdsong recognition using backpropagation and multivariate statistics. *IEEE Trans. Signal Process.* **45**(11), 2740–2748 (1997). doi:10.1109/78.650100
49. Mellinger, D., Martin, S., Morrissey, R., Thomas, L., Yosco, J.: A method for detecting whistles, moans, and other frequency contour sounds. *J. Acoust. Soc. Am.* 4055–4061 (2010). doi:10.1121/1.3531926
50. Mennill, D.J., Burt, J.M., Fristrup, K.M., Vehrencamp, S.L.: Accuracy of an acoustic location system for monitoring the position of duetting songbirds in tropical forest. *J. Acoust. Soc. Am.* **119**, 2832–2839 (2006). doi:10.1121/1.2184988. <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2247711/>
51. Menze, S., Zitterbart, D.P., van Opzeeland, I., Boebel, O.: The influence of sea ice, wind speed and marine mammals on southern ocean ambient sound. *R. Soc. Open Sci.* **4**(1), 160370 (2017). doi:10.1098/rsos.160370. <https://doi.org/10.1098/rsos.160370>
52. Mercado III, E., Sturdy, C.B.: Classifying animal sounds with neural networks. In: Brown, C.H., Riede, T. (eds.) *Comparative Bioacoustics: An Overview*, Chap. 10. Bentham Science Publishers, Oak Park, IL (2016)

53. Montavon, G., Orr, G., Müller, K.R. (eds.): *Neural Networks: Tricks of the Trade*. Springer, New York (2012)
54. Mporas, I., Ganchev, T., Kocsis, O., Fakotakis, N., Jahn, O., Riede, K., Schuchmann, K.L.: Automated acoustic classification of bird species from real-field recordings. In: 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, vol. 1, pp. 778–781. IEEE, New York (2012). doi:10.1109/ICTAI.2012.110
55. Murphy, K.P.: *Machine Learning: A Probabilistic Perspective*. MIT Press, Cambridge (2012)
56. Murphy, K.P., Paskin, M.A.: Linear-time inference in hierarchical HMMs. In: *Advances in Neural Information Processing Systems*, vol. 2, pp. 833–840 (2002)
57. North American Bird Conservation Initiative: *State of North America's birds 2016*. Tech. rep., Environment and Climate Change Canada, Ottawa, Ontario (2016). <http://www.stateofthebirds.org/2016/state-of-the-birds-2016-pdf-download/>
58. Padgham, M.: Reverberation and frequency attenuation in forests—implications for acoustic communication in animals. *J. Acoust. Soc. Am.* **115**, 402 (2004). doi:10.1121/1.1629304
59. Perez, E.C., Fernandez, M.S.A., Griffith, S.C., Vignal, C., Soula, H.A.: Impact of visual contact on vocal interaction dynamics of pair-bonded birds. *Anim. Behav.* **107**, 125–137 (2015). doi:10.1016/j.anbehav.2015.05.019
60. Pieretti, N., Farina, A., Morri, D.: A new methodology to infer the singing activity of an avian community: the Acoustic Complexity Index (ACI). *Ecol. Indic.* **11**(3), 868–873 (2011). doi:10.1016/j.ecolind.2010.11.005
61. Podos, J., Moseley, D.L., Goodwin, S.E., McClure, J., Taft, B.N., Strauss, A.V., Rega-Brodsky, C., Lahti, D.C.: A fine-scale, broadly applicable index of vocal performance: frequency excursion. *Anim. Behav.* **116**, 203–212 (2016). doi:10.1016/j.anbehav.2016.03.036
62. Ptacek, L., Machlica, L., Linhart, P., Jaska, P., Muller, L.: Automatic recognition of bird individuals on an open set using as-is recordings. *Bioacoustics* **25**(1), 55–73 (2016). doi:10.1080/09524622.2015.1089524
63. Ranft, R.: Natural sound archives: past, present and future. *Anais da Academia Brasileira de Ciências* **76**(2), 456–460 (2004). doi:10.1590/S0001-37652004000200041
64. Ren, Y., Johnson, M., Clemins, P., Darre, M., Glaeser, S., Osiejuk, T., Out-Nyarko, E.: A framework for bioacoustic vocalization analysis using hidden Markov models. *Algorithms* **2**(4), 1410–1428 (2009). doi:10.3390/a2041410
65. Ross, J.C., Allen, P.E.: Random forest for improved analysis efficiency in passive acoustic monitoring. *Ecol. Inform.* (2013). doi:10.1016/j.ecoinf.2013.12.002
66. Ruiz-Muñoz, J., You, Z., Raich, R., Fern, X.Z.: Dictionary learning for bioacoustics monitoring with applications to species classification. *J. Signal Process. Syst.* 1–15 (2016). doi:10.1007/s11265-016-1155-0
67. Sandsten, M., Ruse, M.G., Jönsson, M.: Robust feature representation for classification of bird song syllables. *EURASIP J. Adv. Signal Process.* **2016**(1) (2016). doi:10.1186/s13634-016-0365-8. <http://dx.doi.org/10.1186/s13634-016-0365-8>
68. Scott Brandes, T.: Automated sound recording and analysis techniques for bird surveys and conservation. *Bird Conserv. Int.* **18**(S1), 163–173 (2008). doi:10.1017/S0959270908000415
69. Somervuo, P., Härma, A., Fagerlund, S.: Parametric representations of bird sounds for automatic species recognition. *IEEE Trans. Audio Speech Lang. Process.* **14**(6), 2252–2263 (2006). doi:10.1109/TASL.2006.872624
70. Stein, R.C.: Modulation in bird sounds. *The Auk* **85**(2), 229–243 (1968). doi:10.2307/4083583
71. Stoddard, P.K., Owren, M.J.: Filtering in bioacoustics. In: Brown, C.H., Riede, T. (eds.) *Comparative Bioacoustics: An Overview*, Chap. 7. Bentham Science Publishers, Oak Park, IL (2016)
72. Stowell, D., benetos, E., Gill, L.F.: On-bird sound recordings: Automatic acoustic recognition of activities and contexts. *IEEE/ACM Trans. Audio Speech Lang. Process.* **25**(6), 1193–1206 (2017)
73. Stowell, D., Gill, L.F., Clayton, D.: Detailed temporal structure of communication networks in groups of songbirds. *J. R. Soc. Interface* **13**(119) (2016). doi:10.1098/rsif.2016.0296

74. Stowell, D., Plumbley, M.D.: Segregating event streams and noise with a Markov renewal process model. *J. Mach. Learn. Res.* **14**, 1891–1916 (2013). <http://jmlr.org/papers/v14/stowell13a.html>
75. Stowell, D., Plumbley, M.D.: Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. *PeerJ* **2**, e488 (2014). doi:10.7717/peerj.488
76. Stowell, D., Plumbley, M.D.: Large-scale analysis of frequency modulation in birdsong databases. *Methods Ecol. Evol.* (2014). doi:10.1111/2041-210X.12223. <http://arxiv.org/abs/1311.4764>
77. Stowell, D., Wood, M., Stylianou, Y., Glotin, H.: Bird detection in audio: a survey and a challenge. In: *Proceedings of MLSP 2016* (2016)
78. Sueur, J., Farina, A.: Ecoacoustics: the ecological investigation and interpretation of environmental sound. *Biosemiotics* 1–10 (2015). doi:10.1007/s12304-015-9248-x
79. Sueur, J., Farina, A., Gasc, A., Pieretti, N., Pavoine, S.: Acoustic indices for biodiversity assessment and landscape investigation. *Acta Acustica United with Acustica* **100**(4), 772–781 (2014). doi:10.3813/AAA.918757
80. Sueur, J., Pavoine, S., Hamerlynck, O., Duvail, S.: Rapid acoustic survey for biodiversity appraisal. *PLoS One* **3**(12), e4065 (2008). doi:10.1371/journal.pone.0004065
81. Suzuki, R., Matsubayashi, S., Nakadai, K., Okuno, H.G.: Localizing bird songs using an open source robot audition system with a microphone array. In: *Interspeech 2016. International Speech Communication Association* (2016). doi:10.21437/interspeech.2016-782. <http://dx.doi.org/10.21437/Interspeech.2016-782>
82. Tchernichovski, O., Nottebohm, F., Ho, C.E., Pesaran, B., Mitra, P.P.: A procedure for an automated measurement of song similarity. *Anim. Behav.* **59**(6), 1167–1176 (2000). doi:10.1006/anbe.1999.1416
83. Ter Maat, A., Trost, L., Sagunsky, H., Seltmann, S., Gahr, M.: Zebra finch mates use their forebrain song system in unlearned call communication. *PLoS One* **9**(10), e109334 (2014). doi:10.1371/journal.pone.0109334
84. The state of nature in the UK and its overseas territories. Tech. rep., RSPB and 24 other UK organisations (2013). <http://www.rspb.org.uk/ourwork/projects/details/363867-the-state-of-nature-report>
85. Towsey, M., Planitz, B., Nantes, A., Wimmer, J., Roe, P.: A toolbox for animal call recognition. *Bioacoustics* **21**(2), 107–125 (2012). doi:10.1080/09524622.2011.648753
86. Towsey, M., Zhang, L., Cottman-Fields, M., Wimmer, J., Zhang, J., Roe, P.: Visualization of long-duration acoustic recordings of the environment. *Proc. Comput. Sci.* **29**, 703–712 (2014). doi:10.1016/j.procs.2014.05.063. <http://dx.doi.org/10.1016/j.procs.2014.05.063>
87. Vannoni, E., McElligott, A.: Fallow bucks get hoarse: vocal fatigue as a possible signal to conspecifics. *Anim. Behav.* **78**(1), 3–10 (2009). doi:10.1016/j.anbehav.2009.03.015
88. Vannoni, E., McElligott, A.G.: Low frequency groans indicate larger and more dominant fallow deer (*Dama dama*) males. *PLoS One* **3**(9), e3113 (2008). doi:10.1371/journal.pone.0003113
89. Ventura, T.M., de Oliveira, A.G., Ganchev, T.D., de Figueiredo, J.M., Jahn, O., Marques, M.L., Schuchmann, K.L.: Audio parameterization with robust frame selection for improved bird identification. *Expert Syst. Appl.* (2015). doi:10.1016/j.eswa.2015.07.002. <http://dx.doi.org/10.1016/j.eswa.2015.07.002>
90. Vincent, E., Araki, S., Theis, F., Nolte, G., Bofill, P., Sawada, H., Ozerov, A., Gowreesunker, V., Lutter, D., Duong, N.Q.: The signal separation evaluation campaign (2007–2010): achievements and remaining challenges. *Signal Process.* **92**(8), 1928–1936 (2012). doi:10.1016/j.sigpro.2011.10.007
91. Vitter, J.S.: Random sampling with a reservoir. *ACM Trans. Math. Softw.* **11**(1), 37–57 (1985). doi:10.1145/3147.3165. <http://dx.doi.org/10.1145/3147.3165>
92. wa Maina, C., Muchiri, D., Njoroge, P.: A bioacoustic record of a conservancy in the Mount Kenya ecosystem. *Biodivers. Data J.* **4**, e9906 (2016). doi:10.3897/BDJ.4.e9906. <http://dx.doi.org/10.3897/BDJ.4.e9906>

93. Walters, C.L., Freeman, R., Collen, A., Dietz, C., Brock Fenton, M., Jones, G., Obrist, M.K., Puechmaille, S.J., Sattler, T., Siemers, B.M., et al.: A continental-scale tool for acoustic identification of European bats. *J. Appl. Ecol.* **49**, 1064–1074 (2012). doi:10.1111/j.1365-2664.2012.02182.x
94. Webster, M.S., Budney, G.F.: Sound archives and media specimens in the 21st century. In: Brown, C.H., Riede, T. (eds.) *Comparative Bioacoustics: An Overview*, Chap. 11. Bentham Science Publishers, Oak Park, IL (2016)
95. Wichern, G., Xue, J., Thornburg, H., Mechtley, B., Spanias, A.: Segmentation, indexing, and retrieval for environmental and natural sounds. *IEEE Trans. Audio Speech Lang. Process.* **18**(3), 688–707 (2010). doi:10.1109/TASL.2010.2041384
96. Wiley, R.H.: Associations of song properties with habitats for territorial oscine birds of eastern North America. *Am. Nat.* 973–993 (1991)
97. Williams, H., Levin, I., Norris, D., Newman, A., Wheelwright, N.: Three decades of cultural evolution in savannah sparrow songs. *Anim. Behav.* **85** (2013). doi:10.1016/j.anbehav.2012.10.028
98. Wilson, D.R., Ratcliffe, L.M., Mennill, D.J.: Black-capped chickadees, poecile atricapillus, avoid song overlapping: evidence for the acoustic interference hypothesis. *Anim. Behav.* **114**, 219–229 (2016). doi:10.1016/j.anbehav.2016.02.002. <http://dx.doi.org/10.1016/j.anbehav.2016.02.002>
99. Zann, R.A.: *The Zebra Finch: A Synthesis of Field and Laboratory Studies*, vol. 5. Oxford University Press, Oxford (1996)
100. Zuidema, W.: Context-freeness revisited. In: *Proceedings of CogSci 2013* (2013)

Chapter 12

Audio Event Recognition in the Smart Home

Sacha Krstulović

Abstract After giving a brief overview of the relevance and value of deploying automatic audio event recognition (AER) in the smart home market, this chapter reviews three aspects of the productization of AER which are important to consider when developing pathways to impact between fundamental research and “real-world” applicative outlets. In the first section, it is shown that applications introduce a variety of practical constraints which elicit new research topics in the field: clarifying the definition of sound events, thus suggesting interest for the explicit modeling of temporal patterns and interruption; running and evaluating AER in 24/7 sound detection setups, which suggests to recast the problem as open-set recognition; and running AER applications on consumer devices with limited audio quality and computational power, thus triggering interest for scalability and robustness. The second section explores the definition of user experience for AER. After reporting field observations about the ways in which system errors affect user experience, it is proposed to introduce opinion scoring into AER evaluation methodology. Then, the link between standard AER performance metrics and subjective user experience metrics is being explored, and attention is being drawn to the fact that F-score metrics actually mash up the objective evaluation of acoustic discrimination with the subjective choice of an application-dependent operation point. Solutions to the separation of discrimination and calibration in system evaluation are introduced, thus allowing the more explicit separation of acoustic modeling optimization from that of application-dependent user experience. Finally, the last section analyses the ethical and legal issues involved in deploying AER systems which are “listening” at all times into the users’ private space. A review of the key notions underpinning European data and privacy protection laws, questioning if and when these apply to audio data, suggests a set of guidelines which summarize into empowering users to consent by fully informing them about the use of their data, as well as taking reasonable information security measures to protect users’ personal data.

S. Krstulović (✉)
Audio Analytic Ltd., 2 Quayside, Cambridge CB5 8AB, UK
e-mail: sacha.krstulovic@audioanalytic.com

Keywords Smart home applications • Audio event recognition • Modelling audio events • Computational power • Embedded sound recognition • Audio quality • Open-set recognition • User interface • Objective and subjective evaluation • Ethics • Privacy

12.1 Introduction

Progress in IP networking and the miniaturization of network chips has made it possible to connect virtually any object to the Internet, thus enabling new services and value. This gave birth to a new concept, and related global market, known as “the Internet of things” (IoT). As this market structured itself [6], distinctive application segments have emerged such as, e.g., smart cities or wearables. The aim of this chapter is to explore the applications of audio event recognition (AER)¹ into the IoT market segment known as the *smart home* or *connected home* [1, 53, 72], as well as to analyze how applications in this market may inform AER research and evaluation methods, thus establishing *pathways to impact* from AER research to applications.

What are Smart Home Applications? Smart home applications, also known as home automation and formerly known as domotics, aim at making use of technology to provide comfort, convenience, security, and entertainment to the home’s inhabitants [2]. These applications can be divided into *time saving* applications, where certain tasks are automated to gain time and comfort, and *time using* applications, related to entertainment. Their evolution closely follows that of more generic technological advances [2]. As a matter of fact, it is the general availability of electricity in the early twentieth century which allowed homes to be equipped with, e.g., washing machines and vacuum cleaners, a form of automation of common household tasks, as well as TV sets, then a new form of entertainment. In the early eighties, the availability of microprocessors triggered attempts at marketing what was then branded as domotics, where a central processor would react to simple sensors and trigger certain programmed actions, from simple ones such as modulating the home’s heating system, to more far-fetched concepts such as pouring a hot bath automatically upon detection of presence in the bathroom at certain times of the day. Domotics at that time met limited success, due to the need for costly wiring, and the lack of a universal communication protocol between appliances. These blockers were subsequently lifted by three technological advances: Internet protocol (IP) communication in the early eighties, wireless Internet technology in

¹The term audio event recognition (AER) in this chapter corresponds to what is referred to as sound event detection in other chapters of this book. Whereas consensus is currently forming amongst the academic research community around the latter term, the industry prefers AER for marketing reasons: firstly because it establishes a parallel with automatic speech recognition, and secondly because “recognition” makes the system feel more intelligent by referring to semantics and meaning, than “detection” which refers to “plain automation.”

the nineties, and more recently the miniaturization of wireless communication chips. Nowadays, almost any device can be equipped with such wireless communication, leading to the concept known as the Internet of things [30].

The Rise of AI in the Smart Home One further and most recent technological advance is that of artificial intelligence (AI). Although the concept of AI covers a broad range of computer science techniques, one of its manifestations most visible to the general public is the level of maturity and usability achieved by voice interfaces, i.e., automatic speech recognition, speech synthesis, and artificial dialog systems. While this maturation was first supported by the mobile phone industry and later on by in-car hands-free applications, a new class of voice operated *smart home assistant* appliances are also appearing, as exemplified by the Amazon Echo and Google Home products [29]. The interesting point here is that via these devices, AI applied in the audio domain has become a key driver of the smart home market [29].

How Does AER Fit Into Smart Home Applications? Li et al. mention three generations of home automation technologies [39]:

1. the *wireless technology and proxy server* approach, where sensors are used to monitor the occupant's activities and report to a server in charge of operating other electrical devices according to pre-defined programs;
2. the *artificial intelligence* (AI) approach, where a larger proportion of adaptiveness is introduced by resorting to AI methods to learn the occupant's behavioral patterns and adapt the programs automatically to changes in behavior, for greater flexibility;
3. the *robot buddy*, where the system's user interface and behavior aim at implementing a basic form of personality into the system, to make it more user friendly.

Audio event recognition feeds in various ways into these three generations of smart home applications:

1. at the simplest level, AER is able to supply a sophisticated sensing modality, by flagging the presence of certain sounds or recognizing the occurrence of certain scenes, beyond mere acoustic loudness measurements;
2. at the next level, it allows access to a semantic interpretation of what is happening into the home, for example, the sound of a smoke alarm indicating the probability of a fire;
3. in the case of an embodied home assistant, AER materializes the human-like functions of hearing and listening, beyond the sounds of speech and music.

From a marketing perspective, Audio Event Recognition in the Smart Home thus unlocks several types of value.

- *Peace of mind*: is something bad happening when I am away from home?
- *Intelligent audio sensing*: giving the home a form of smartness by enabling the artificial awareness of what is happening, via machine listening.
- *Improved audio sensing performance*: AER aims at modeling and recognizing complex audio patterns. Thus, it yields better false alarm rates than mere acoustic

level sensing for applications which require the distinctive targeting of particular audio events.

- *Complementarity of sensing modalities*: compared to video, sounds can be sensed in the dark and around corners.
- *Aggregation of sensing modalities*: AER can help disambiguating other sensors, e.g., switching off motion sensing if dog sounds have been detected in the home.

Thus AER can deliver lots of value in the context of smart home applications, some of it unique to the sound sensing modality.

Examples of AER Applications for the Smart Home Success in the commercialization of a piece of technology depends on more than algorithmic performance: defining products and use cases is crucial, as it connects novel technical capabilities with end user needs and aspirations, thus justifying purchase of the technology and economical impact. Looking at Audio Analytic's range of AER-based products, branded ai3™, gives an idea of a range of possible applications and use cases:

Window Break A window is broken in an unoccupied home. The sound is recognized by the sound recognition system and the home's HiFi system automatically plays loud music or pre-recorded sounds to deter any intruder. Lights in the house are automatically switched on. An alert is sent to the home owner's phone so that they can check their home cameras and if necessary alert the authorities.

Smoke&CO Alarms An alarm sounds in an unoccupied home. Regardless of the tone pattern of the alarm or the number of devices sounding in the home at once, the sound is recognized by the sound recognition system, and an alert is sent to the home owner's phone. If an alarm sounds at night when a family is asleep, the sound recognition system recognizes the sound and lights automatically switch on to aid a safe and rapid exit.

Baby Cry A baby stirs and begins to cry at night. The sound is recognized by the sound recognition system and the home sound system automatically plays a soothing lullaby to help baby back to sleep. If baby keeps crying for a given period of time, the sound recognition system sends an alert to one of the parent's wearable device, waking one parent without disturbing the other. The sound recognition system connects to the home lighting system turning on night lights so mum or dad can make their way to the nursery safely.

Dog Bark A dog alone at home begins to bark. The sound is recognized by the sound recognition system and an alert is sent to the owner's mobile device. There are two primary use cases for dog bark: security and pet care. A dog barking can be an early warning sign of a potential intruder. The owner can choose to view the home camera feed to check the property is secure, trigger lights to come on. Alternatively, they can speak directly to the pet via the intercom and dispense snacks from the automated feeding station.

Anomaly Detection the sound recognition system can establish the normative sound profile for an individual home. Anomalies can include sounds such as aggressive

shouting, calls for help, forced entry—or other anomalies in the home. An anomaly in the sound profile of the home is recognized by the sound recognition system. An alert is sent to the home owner or nominated carer so they can take appropriate action.

Custom Sounds the sound recognition system can be programmed by the user to recognize specific sounds around the home. Typical sounds a user might wish to be alerted to include intruder alarms, white goods alerts (e.g., washing machine end-of-cycle beeps, microwave oven beeps, etc.) or the sound of a doorbell. Once trained to recognize the sound, the sound recognition system can trigger automated responses such as message alerts to the home owner or connected device actions.

These are not the only possible use cases for AER, since professional security, industrial machine surveillance, elderly care, smart cities and more can be imagined as possible outlets. However, Audio Analytic has found smart home applications to be the ones yielding the most dynamic and sizable commercial traction amongst the wide variety of Internet of things sub-domains.

Relationship Between Academic Research and Industrial R&D Defining and analyzing use cases for AER technology is an interesting element of the relationship between academic research and industrial R&D. On the one hand, academic research is expected to generate the fundamental knowledge which will support the potential for future applications. It is generally accepted that the development of scientific knowledge should not be restricted on the grounds of practical limitation, e.g., an algorithm which requires some impractical amount of computation to generate sound recognition improvements still contributes to sound recognition science, perhaps as an intermediate step towards something more tractable in the future. Another difference from the industry is that academic research may not always receive an amount of resourcing that would match the size of industrial investment. For example, academic entities may struggle to get enough funding to realize large data collections. Industrial R&D, on the other hand, is expected to deliver research outcomes which are narrowly focused on improving the practical performance of a viable product, as a matter of minding the return on private investment. From there, the industry often looks at the academic body of work as a reservoir of ideas which can be evaluated as solutions to practical application challenges, then refined and complemented by an extra body of in-house, proprietary research and development. Or, as we like to define it at Audio Analytic, “state-of-the-art, but with our own extra twist.” In the industrial context, the R&D process thus takes an iterative form which oscillates between (a) performing system evaluation across field data, in order to study and define what practical challenges and performance targets need to be solved in order to achieve a viable product, and (b) developing solutions to the problems uncovered as the next wave of improvements, either by channeling academic knowledge, itself often requiring significant modifications, or by developing in-house solutions from the ground up, but with a view on “avoiding to reinvent the wheel.” This requires to be creative but also to be well connected, informed, and able to follow up with the academic state of the art.

From that standpoint, knowledge transfer could be thought of as flowing unidirectionally from the academia to the industry, through research contracts, internships of early stage researchers, employment pipelines and more. Sometimes, the deal can take the form of “knowledge against access to large data sets.” However, there is a possibly more interesting way to conceive the relationship between industrial applications and academic research: without restricting the freedom of academic research, industrial applications can help steering academia towards *useful problems*, for which the scientific generality is not lost, but the pathways to impact are more clearly defined. Examples of this mindset have punctuated the history of automatic speech recognition (ASR) research, e.g., with the fact that automatic dictation, then meeting room transcription [31], then hands-free speech in cars, and nowadays smart home assistants, have successively taken turns in creating traction for the ASR domain by, respectively, surfacing the useful problems of connected speech recognition, then speaker turns and naturally spoken speech, then robustness to background noise, then far-field speech capture, themselves sparking research interest into better language models, speaker diarization and array processing, to name but a few. Thus, it may be important for the AER research community to achieve a similar consensus around a small number of well-defined applications, perhaps contributed by the industry, which could pull the domain forward in a similar way, rather than “shooting in all directions.”

Such reference applications do not have to be smart home applications; however, smart home applications are a good candidate. Indeed, their commercial dynamism may help justifying the pathways to impact, as an asset to obtain research funding. Focusing on the smart home would imply design choices for AER research, for example, that of working with indoors sounds and room acoustic constraints, but this does not reduce the generality of the AER research challenge, insofar as home sounds still include a number of acoustic classes (e.g., resonances, percussions, beeps, etc.) and acoustic phenomena (e.g., polyphony, reverberation, channel distortions, etc.) whose variety is as wide as for other applications. Or in other terms, dealing with smart home sounds can help focusing the research and justifying its impact, without reducing the general nature of the AER problem when studied within the context of this particular class of applications. To a large extent, Sects. 12.2 and 12.3 of this chapter were precisely written with the mindset of illustrating how smart home applications can contribute such useful problems to the research community, both in the area of acoustic modeling and in the area of systems evaluation.

Outline of this Chapter As a summary, smart home applications suggest certain requirements on what the technology is expected to deliver. One of the goals of this book chapter is precisely to analyze how smart home applications can inform AER research methodology and research topics, in order to maximize the usefulness of research and to optimize its pathways to impact. With this in view, the chapter explores three aspects of the development and productization of AER for the smart home. Section 12.2 analyses novel research directions elicited by the practical constraints of deploying AER across real smart home devices, in particular when

it comes to achieving a precise definition of sound events, evaluating the system in a 24/7 listening setup, and running the system on consumer products with limited audio capture quality and limited computational power. Section 12.3 looks at user experience questions, by exploring the nature of user experience for AER, proposing subjective performance metrics, and analyzing the gap which may still exist between today's standard AER performance metrics and the need to optimize user experience. Finally, Sect. 12.4 reviews the ethical and privacy protection issues posed by the 24/7 processing of private audio data, which in themselves play an important part into the general public's perception of AER technology.

12.2 Novel Research Directions Elicited by AER Applications in the Smart Home

This section discusses three aspects in play when deploying an AER system into smart home applications: the necessity to redefine sound events as interrupted units, recasting 24/7 recognition as an open-set problem, and the constraints imposed by imperfect sound capture and finite computational power. In relation to these topics, new research directions or evaluation practices are being proposed.

12.2.1 *Audio Events as Structured Interrupted Sequences*

Chapter 8 of this book, entitled “approaches to complex sound scene analysis”, mentions that many of the detection methods that can be encountered in the literature produce frame-by-frame outputs of event presence/absence, with decisions sometimes taken globally across a longer observation buffer, according to a classification principle akin to the bag of frames (BoF) approach [3, 61]. However, what users define as sound events may not exactly correspond to a continuous series of consistent audio frames. For example, in the cases of intermittent baby cries or smoke alarm patterns, the target sound is interleaved with silence or background noise. Furthermore, what may define an audio event of interest for users may not be pertaining solely to local acoustic patches: what may distinguish, say, a smoke alarm from an alarm clock may not so much be a short bag of time-frequency atoms than it may be some longer term sequential characteristics, such as the normalized T3/T4 beeping patterns [35]. Taking the example of baby cries, there may be a typical length of crying related to babies' average lung capacity. Furthermore, users may be interested in alerts about long episodes of crying rather than isolated screams, or may want to receive a single alert after the smoke alarm has sounded for 10 s versus one alert for every beep or every matching audio frame.

The idea here is that users hear concepts, not acoustic frames: what users define as a sound event may actually correspond to a long term, impure yet temporally

structured sequence, whereas the bag of frame approach mostly assumes some form of consistency between all the frames in the bag, for example, all the frames coming from a given musical instrument or from a given audioscape. By definition, the BoF approach discards the modeling of long-term temporal structure or interruption.

Chapters 5 and 8 report several research directions which aim at introducing temporal modeling in AER, for example, hidden Markov models (HMMs), explicit duration HMMs, score post-processing or various forms of recurrence or temporal context models in deep neural networks. But when it comes to the modeling of interruption, although polyphonic event detection techniques, also exposed in Chap. 8, may bring a solution to the problem of modeling interleaved audio classes, the above-mentioned approaches may not fully address the explicit modeling of *long-term* temporal patterns (of the order of several seconds), or the potentially very wide acoustic variability of the “in between” frames (e.g., the gaps between smoke alarm beeps, where anything could happen). Both problems of impurity and temporal modeling are related, insofar as the “in between” frames may not be good predictors for the parts of the sequences which are actually of interest. Thus, if thinking of standard implicit sequence predictors such as hidden Markov models or recurrent neural networks, what happens in the gaps may not be a good predictor, e.g., of the smoke alarm beeps or the baby cries themselves.

Furthermore, this ambiguity in the definition of sound events manifests itself as a dilemma for standard evaluation metrics: in a 24/7 AER framework, target audio events appear as series of variable length blobs across a continuum of background sounds, and there does not seem to be a definitive consensus achieved on what the right level of granularity should be for the definition of the countable acoustic units used in F-score, precision, and recall [43] (see also Chap. 6). At one extreme, counting whole events as recognition units may bias the evaluation: as a thought exercise, a 2 h movie sound may have a higher chance to trigger a baby cry false alarm than a 5 s cat meow, because of their difference in complexity, length, and coverage. At the other extreme, counting frames as classification units misses out on the longer term temporal modeling. In between these extremes, counting blocs of a finite length [43] suffers from the problem of the variable presence of non-target frames into the “arbitrary” acoustic unit. The same problem of granularity and impurity also appears in weak labeling [28], where the contents of constant length audio chunks is labeled rather than precise event boundaries.

The idea here is that there is a need for a richer definition of sound events, away from the bag of frames, continuous acoustics, or “beads on a string” [50] mindsets inherited from speech, speaker, or music recognition. The more explicit modeling of inconsistent interruptions and long-term temporal patterns could find inspiration from, e.g., automatic speech synthesis techniques. Indeed, in this domain, multi-space densities [66] have been used to model the acoustic inconsistency between the voiced and unvoiced parts of speech. Furthermore, the user’s definition of audio events for practical application brings support to the suggestions made in Chap. 8 to resort to temporal modeling more explicitly. There again, additional inspiration may come from the speech synthesis domain, where explicit duration densities [79, 81] or hidden semi-Markov models (HSMMs) [46, 80] have already been used to

learn typical phoneme durations from data, instead of relying on implicit sequential consistency. Solutions may also come from a notion of n-grams or multigrams [23] similar in spirit to the language models used in the early days of speech recognition. While results of applying such techniques to the AER domain remain to be more widely published, the hope of this section is to trigger a change of mindset away from the definition of sound events as bags of instantaneous acoustic frames and higher up from frame-by-frame classification machines, towards the wider scope of modeling sound events as interrupted acoustic sequences, in a way that would integrate instantaneous acoustic modeling and wider sense temporal and structural models more tightly.²

12.2.2 *Continuous 24/7 Sound Recognition as an Open-Set Problem*

System evaluation in past scientific publications has often de facto reduced AER to a closed set problem, by reporting experiments involving a limited number of sound classes. For example, comparative evaluation campaigns such as DCASE 2013 [58] or CLEAR [62], respectively, involved 16 and 13 sound classes, thus leading to investigate 16×16 or 13×13 -sized confusion matrices. However, an applied AER system actually listens 24/7. In this context, in contrast to speech recognition systems, it makes no sense for an AER system to have a wake-up button (e.g., long press on a mobile phone's home button, or voice listening button on a steering wheel) or to be triggered by a keyword (e.g., "Alexa" or "OK Google" to wake up the system's speech recognition function).

Although the set of non-target sounds can be thought of as somewhat bounded by the target environment (e.g., a home indoors would most likely exclude loud car engine noises), it remains difficult to enumerate the interfering non-target sound classes exhaustively, i.e., more difficult than enumerating the phonemes of speech or the notes of musical instruments. Thus, the confusion matrix for a real AER problem might be better formalized as $1 \times$ near infinity, a case known as *open-set recognition* [56]. Progress on this has been achieved in the domain of image recognition. The underlying theory introduces a general notion of open-space risk coming as a complement to the notion of empirical risk which supports the

²This suggestion may be reminiscent of speech recognition techniques, where the acoustic models and the language model contribute almost equally to speech recognition accuracy [54]. However, the problem may be different in AER: the proportion of silence or interruption versus target acoustic frames may be much smaller, and thus have less effect on the general acoustic probabilities, in continuous speech than in the case of short interrupted audio events such as, e.g., smoke alarms or baby cries. Besides, the structure of non-speech audio events or audio scenes may not be of a linguistic nature according to the strict definition of language as a system of communication, thus questioning the structural nature of non-speech audio events at a deeper level of cognitive concepts. More discussion on "acoustic language models" can be found in Chap. 8.

standard posterior probability models. The minimization of empirical risk is thus balanced and regulated by some extra optimization stages related to minimizing the open-set risk. Concretely, this idea has been applied in [56] to extend binary support vector machines (SVMs) and 1-class SVMs to “binary 1-vs-set” machines, with significant improvements reported in terms of generalization and rejection of completely unseen classes. The development of open-set deep neural networks (DNNs) for image classification has also been reported more recently in [5], with successful results in terms of resistance to adversarial images, as well as detection of unknown classes. Early results of applying open-set methodology to audio scene classification have been reported in [4], where a particular type of 1-class SVM has shown promising generalization results across two public data sets which were limited in size. Such results remain to be extended to audio event recognition rather than audio scene recognition, testing over larger data sets, and possibly developing other open-set machines than those derived from 1-class SVMs.

Recasting the 24/7 AER problem as an open-set problem underlines some of the limitations of the current evaluation practice of using F-score, precision, recall and related metrics (e.g., area under curve, see Chap. 6 for more definitions):

1. these metrics highly depend on the composition and balance of the considered test set,
2. they assume that sounds are identifiable positive or negative units, whereas non-target sounds may actually be a continuum in a 24/7 sound recognition context.

Indeed, in the 24/7 sound recognition context the system becomes exposed across time to a growing amount of non-target sounds, with the prior probabilities on non-target sounds tending to one and the probabilities of target sounds tending to zero. Assuming that the amount of false positives is proportional to the exposure to non-targets, this means that the precision, defined as $\frac{TP}{TP+FP}$ where TP and FP denote true positives and false positives, respectively, will tend to zero across time, without anything having changed into the AER system itself. Similarly, the F-score, defined as:

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = 2 \cdot \frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}$$

where recall is defined as $\frac{TP}{TP+FN}$ and FN means false negatives, will tend to zero as the system’s exposure to non-targets tends to infinity much faster than the system’s exposure to target sounds. System evaluation is thus stuck into a dilemma:

- either considering a test set including a balanced number of target and non-target classes, by assuming or designing some adequate way of down-sampling the non-target classes uniformly,
- or considering a test set where the size and coverage of the non-target sounds is significantly larger than the coverage of the target sounds, in an attempt to model the huge priors on the quasi-infinite amount of non-targets.

The dilemma here is that although both approaches may be valid in principle, they are likely to lead to different precision and F-score figures, whereas the classification system itself has not changed. This implies that precision, recall, and F-score *report information about the dataset itself*, in addition to information about the system. Although these evaluation metrics remain appropriate for comparison over a fixed data set, they become difficult to interpret as absolute indicators of system performance in the context of 24/7 sound detection, thus suggesting that approaching 24/7 detection as an open-set problem will require to redefine the AER evaluation metrics.

One preliminary solution to that problem is to resort to false alarm rates per unit of time rather than absolute false alarm counts. For example, the usage of detection error trade-off (DET) curves [41] involving false alarm rates per time unit rather than absolute FA rates has been published, e.g., in the context of keyword spotting in [74]. DET curves themselves have another advantage over F-scores, that of separating discrimination from calibration, which will be discussed in more detail in Sect. 12.3.3.

12.2.3 “The Real World”: Coping with Limited Audio Quality and Computational Power

In the context of smart home applications, AER has to run on consumer grade electronics, and products have to be kept within an acceptable price range, a proportion of which depends on the bill of materials for microphones, codec chips, and computation processing units. Coping with such “imperfections” is actually essential to applicative success, and emphasizes a need to focus on some research directions which are currently understudied in the AER field.

Variations and Limitations of Audio Capture Channels In contrast to mobile phones, where audio quality is deemed crucial to the perceived overall quality of the product, many smart home devices offer only limited audio capture quality. Audio device quality is indeed quite polarized between two extremes: on the one hand, the most common smart home devices will have been designed to deliver 16 kHz, mono, 16 bits digital audio, captured by MEMS or electret microphones of medium quality which have large distortion tolerances, and backed up by low-cost digital codec chips. Meanwhile, at the pricier end, smart home assistant appliances such as Amazon Echo or Google Home have started introducing microphone arrays backed up by powerful digital signal processing (DSP) hardware accelerated subsystems. The latter came with the proof that array processing and source separation were essential to the success of far-field speech recognition.

This has not yet happened in the field of AER, as many customers simply require AER to run within the limitations of 16 kHz / 16 bits mono channels, and within the tolerances of low-cost audio subsystems. Thus, one of the challenges of AER research is to come up with methods and algorithms which are robust against low

quality audio, the presence of channel distortions and the presence of additive noise. While standard noise reduction techniques may work against simple additive white noise, the challenges may be harder to solve for more complex and less stationary distortions such as, e.g., electro-magnetic interference (EMI) noise, or interferences related to digital communications over Wi-Fi.

Also, it should be noted that existing far field audio enhancement methods are often geared towards speech signals, by including prior knowledge specific to the acoustic nature of speech (e.g., [47]). Generalizing such audio enhancement methods to the far-field capture of other sounds than speech remains an understudied research topic.

The above considerations suggest the following points:

- From the standpoint of pathways to impact, it would be useful to identify channel robustness more clearly as a topic of interest for AER research, just like it has been a topic of interest for decades in the context of speech and speaker recognition research.
- Interest for robustness uncovers a limitation of most publicly available data sets, where the lack of either channel identification or channel variety precludes the design and evaluation of methods geared towards coping with channel variability.
- The methods which are currently known to improve far field speech recognition may or may not transpose gracefully to AER, due to the difference of acoustic nature between speech and a more generic set of environmental sounds (including, e.g., percussive sounds).

More generally, whereas AER research has so far focused a lot on the problem of polyphony and source separation, the suggestion here is that a wider diversity of topics related to audio quality, such as channel robustness for AER and the far field capture of environmental sounds, may be worth identifying as salient research topics.

Limited Computational Power and Memory Capacity Whereas a vast proportion of applications in speech recognition, music classification and audio indexing are supported by PC platforms, cloud computing and/or powerful smart phones, most smart home applications are embedded into hardware product whose computational power cannot be expected to match that of a PC. Indeed, IoT devices can be broadly divided into two classes: (a) devices which are perceived as “doing only one thing,” thus requiring the use of low-cost processors to hit a price point that users are willing to pay for what the device does, or (b) embedded devices where sound recognition comes as a bolt-on to add value to an existing product, for example, adding AER capabilities to a consumer grade camera or to a set-top box, thus requiring the algorithm to fit into the device’s existing design and price points. These two cases rule out the use of higher end processors.

Generally speaking, the following features jointly define the financial cost of a processor and the level of constraint imposed on embedded computing:

- the *clock speed* is related to energy consumption;
- the *instruction set* is related to chip size and manufacturing costs, but in some cases includes special instruction sets to parallelize more operations into a single clock cycle;
- the *architecture* defines, e.g., the number of registers, number of cores, presence/absence of a floating point unit (FPU), a graphical processing unit (GPU) and/or a digital signal processing (DSP) unit.

The above features affect applications in the obvious way of defining an upper bound on the number and type of operations which can be executed in a given amount of time, thus ruling the possibility to achieve real-time audio processing at the proportion of processor load allocated to the AER application. In addition, *on-board memory size* is an important factor related to processor cost, as it affects both the computational performance, where repetitive operations can be cached to trade speed against memory, and the scalability of an algorithm, by imposing upper limits on the number of model parameters that can be stored and manipulated.

If trying to work within these limitations, and given that most IoT embedded devices allow Internet connectivity, it could be argued that cloud computing can solve the computational power constraints by abstracting the computing platform and making it virtually as powerful as needed. However, a number of additional design considerations may rule out the use of cloud computing for AESR applications:

- the *latency* introduced by cloud communications can be a problem for, e.g., time critical security applications [9];
- regarding *quality of service* (QoS), network interruptions may introduce an extra point of failure into the system;
- regarding *bandwidth* consumption, sending alerts rather than streaming audio or acoustic features out of the sound recognition device requires less bandwidth;
- last but not least, the continuous streaming of smart home audio to a cloud platform would cause serious *privacy concerns*, whereas running on the device rules out any possibility of eavesdropping [42].

Thus, the reality of embedded industrial applications is that at the price points acceptable into the marketplace, the majority of IoT devices will be devoid of an FPU, will operate in the hundreds of megahertz clock speed range, will not offer on-board DSP or specialized instruction sets, and may prefer to run AER on board rather than in the cloud.

Introducing Computational Cost into the Evaluation In spite of the above-mentioned computational constraints, most research works seem to be producing AER performance evaluations with only limited interest for the computational cost involved: experimental results are most often obtained with floating point arithmetics on powerful computing platforms, and under the assumption that sufficient computing power will sooner or later become available for the algorithm to be practicable in the context of commercial applications. However, a methodology

which disregards computational costs may lead to a roadblock further down the line of pathways to impact. Indeed, the concrete limitations which were depicted in this section suggest that the practicability of state-of-the-art AER algorithms is far from being granted when it comes to running on an average consumer electronics product. In turn, this suggests that the evaluation of AER algorithms at the research level may want to include considerations of computational cost more rigorously as part of the algorithms' evaluation criteria, in a context where justifications of the usefulness of research and clear statements of pathways to impact may be required to obtain research funding, but without venturing too far into spending research resources on porting the algorithm into an actual product, and without limiting the research options on the grounds of practicability.

Our suggestion is that a good balance in this area can be achieved at the research level by evaluating AER accuracy as a function of computational cost. Research results on that topic have been published in [57], where the performances of three types of classifiers commonly used for AER, namely Gaussian mixture models (GMMs) [55], support vector machines (SVMs) [13] and various flavors of deep neural networks (DNNs) [38], are compared as a function of their computational cost over two AER tasks, namely the detection of baby cries and the detection of smoke alarms against a large number of impostor sounds. Such comparison between various types of acoustic models can be tricky, because the results depend to a large extent on the nature of the data set used for the experiments. As a matter of fact, the general consensus [73] is that DNNs require large amounts of data to outperform a GMM or a SVM, or that SVMs tend to outperform other models on small data sets. From that standpoint, the study in [57] can be considered a "fair" comparison insofar as the used data set covers real use cases more closely than previously available data sets and is large enough to avoid imposing a practical handicap on DNNs artificially.

The results, depicted in Fig. 12.1, suggest that GMMs provide a low-cost baseline for classification across both data sets of Baby cry and Smoke Alarms. The GMM acoustic models are able to perform reasonably well at a modest computational cost. SVMs with linear and sigmoid kernels yield similar EER performance compared to GMMs, but their computational cost is overall higher. The computational cost of the SVM is determined by the number of support vectors. Unlike GMMs, SVMs are non-parametric models which do not allow the direct specification of model parameters, although the number of support vectors can be indirectly controlled with regularization. Finally, the results suggest that deep neural networks consistently outperform both the GMMs and the SVMs on both data sets. The computational cost of DNNs can be controlled by limiting the number of hidden units and the number of layers. While changes in the number of units in the hidden layers do not appear to have a large impact on performance, deeper networks appear to perform better in all cases. Additionally, neural networks with ReLU activations achieve good performance, while being an attractive choice for deployment on embedded devices because they do not require expensive look-up table (LUT) operations.

Beyond mere competition between the various compared models, the *methodology* introduced in [57] is one of the important points, which is summarized visually

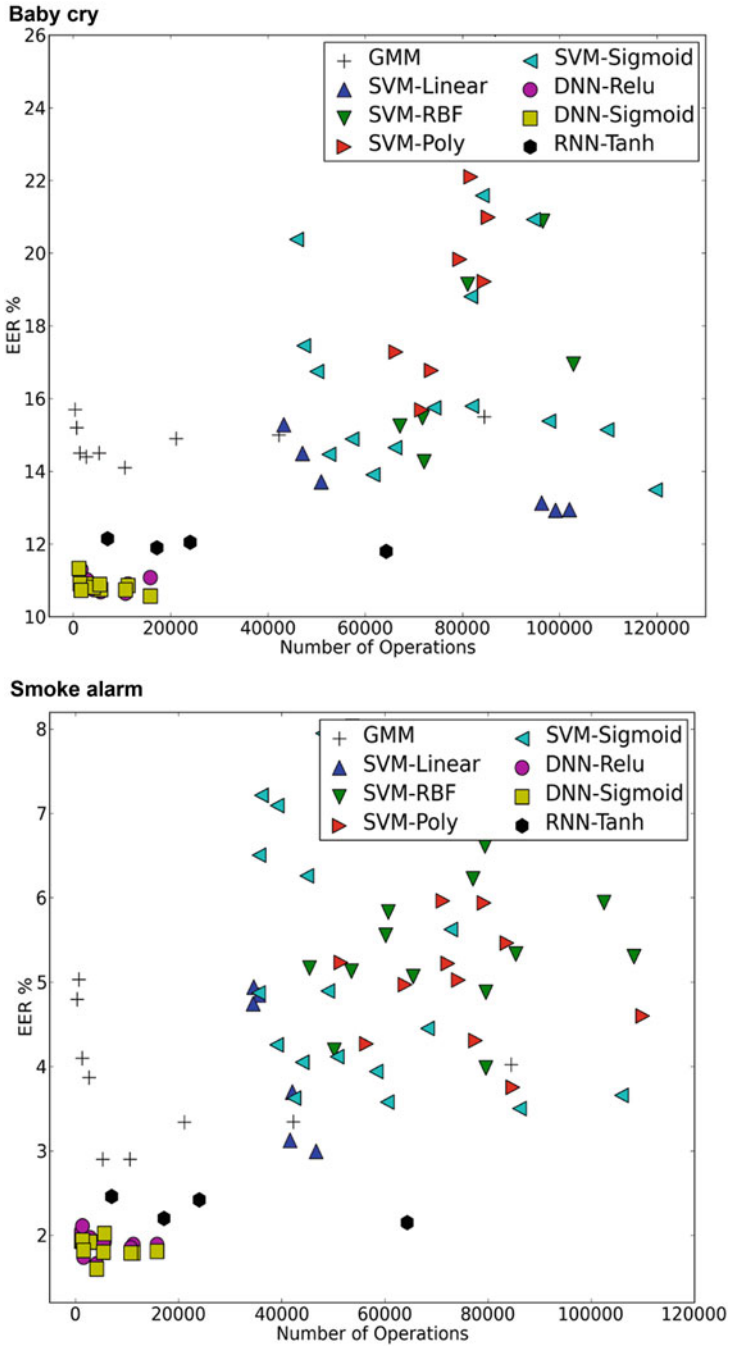


Fig. 12.1 Acoustic frame classification performance (equal error rate percentage, EER) as a function of the number of operations per frame, for each of the tested models, across the baby cry and smoke alarms data sets (from [57]). (top) Baby cry data set. (bottom) Smoke alarm data set

by Fig. 12.1: it is useful to map performance against computational cost in order to choose which algorithm to implement into a particular embedded platform, given the limited computational budget, and after realizing that the computational cost may vary widely between and within particular classes of algorithms. In a similar way, it is useful to investigate the performance of acoustic models as a function of the memory efficiency, since memory is an important consideration when designing models for embedded hardware. Other algorithms than the usual GMMs, DNNs, and SVMs could also be assessed in the same way, and new research methods devised, with the goal of incorporating more explicitly the optimization of computational cost and memory requirements into algorithmic design and structure (e.g., [21]).

12.3 User Experience

At the outlet of *pathways to impact*, applying AER algorithms is expected to deliver value to a user. Productizing AER thus involves users as a new stakeholder in addition to researchers and algorithms. Users may not be educated in AER, may have certain expectations about the system, or certain spontaneous opinions about what the system does or should do. One relevant question is therefore the following: are standard AER evaluation methods good predictors of user experience, user opinion and thus system usefulness? This section explores such user experience aspects.

12.3.1 User Interface Aspects

The fundamental problem behind predicting user experience relates to the understanding of how users interface with an AER system. Looking at automatic speech recognition (ASR), the goal of ASR is to deliver the written transcription of a spoken message: in the context of dictation, users can directly judge whether the transcription returned by the service is consistent with what they have spoken into the microphone, simply by reading the returned text. Thus, word error rate (WER) metrics are by definition correlated with user experience and user opinion. Now looking at automatic text-to-speech synthesis (TTS), user experience depends on the naturalness and the intelligibility of the synthesized speech, two characteristics for which automatic metrics [44, 45] have not been able to correlate well enough with organic user opinions to fully replace human listening tests. Thus, TTS evaluation still largely relies on opinion scoring techniques [16], where a focus group of users actually listens to and rates the system, sometimes with the help of crowdsourcing platforms [12, 24]. From these parallels, two questions arise:

- Through which interface(s) are the results of AER experienced by the user? Or in other words, what is the AER equivalent of reading a text transcription in the case of ASR or listening to a voice in the case of TTS?
- Are there one or several objective metrics which correlate well with the quality of the experience that users may get through the identified interface?

In the case of 24/7 AER systems, the goal is most often to alert the user of the presence or absence of certain sounds into audio recordings. The alert may come with a time stamp indicating when the target sound happened, but that is not mandatory. The value of most AER systems also relates to the notion of *displacement* in a linguistic sense [32, 77], i.e., the capacity of human beings to conceive things which are happening elsewhere in space or time, for example, the occurrence of a particular event at home when the user is away from home, or the occurrence of some event of interest in the past rather than in the present. As such, *remote* alerting has more value than the plain confirmation of the presence of a sound that the user has already heard. This is in contrast, e.g., with ASR, where the users know what they have spoken into the system, or TTS, where the user will to understand the system is saying here and now. In that sense, user experience of AER is more complex to define than user experience of ASR or TTS. As a matter of fact:

- User experience of AER depends to a large extent on the reliability of the remote alerting system. For example, if alerts are not transmitted due to a flaw of the underlying messaging platform, users will think that the sound recognition system as a whole is failing, regardless of the accuracy of the underlying AER algorithm.
- Even in the hypothetical or simulated case where the alerting system would be flawless, checking the truthfulness of a remote audio alert assumes the possibility of double-checking that there was a reasonable cause for the audio alert.

The second point can be solved either directly by allowing the user to listen to a sound snippet which would bracket the audio deemed to have triggered the detection, or indirectly by allowing the user to cross-check that there is a cause through other sensing modalities, e.g., checking a home surveillance camera's video feed for some visible audio source (presence of people, pet, moving objects, etc.). The indirect case is less reliable than the direct one, since the sensor acting as a proxy may not be consistent with audio capture, e.g., the sound source might be out of the camera's field of vision, or occurring in the dark.³

From the standpoint of being able to check audio event alerts against audio snippets, using a true positive or a false positive rate (or the system's precision measure introduced in Chap. 6) would appear a reasonable way to measure user experience. But two additional considerations then come into play:

³A requirement to be able to cross-check the triggering audio may seem to contradict the privacy and eavesdropping concerns analyzed further down in Sect. 12.4.4. However, in practice there is less of a privacy concern about transmitting short audio snippets, with pre and post bracketing time kept to a minimum around the triggering audio event, than there is about streaming someone's personal audio to the cloud in a continuous 24/7 manner.

- What about missed detections? In what sense do they participate to user experience?
- Is there room for subjectivity in any of the considered metrics?

12.3.2 *The Significance of Subjectivity in AER Evaluation*

Type of System Errors and their Impact on User Opinion As introduced in Chap. 6, the system can make two types of errors: false positives, a.k.a. false alarms, defined as sending an alert for a sound which is not the desired target sound, or false negatives, a.k.a. missed detections, defined as not sending an alert when the target sound was actually there.

Missed detections are tricky insofar as they are difficult to cross-check by users. By definition they can only remain un-notified, thus suggesting low impact on user experience. However, the possibility of incidental discovery of a missed detection exists, e.g., discovering a broken window at home while no alert was sent, or being at home while the smoke alarms sounds and seeing no alert coming. Such a discovery could have a disastrous impact on user opinion, and could instantly void the AER system's credibility.

This brings about the question of subjectivity. In the case of ASR transcriptions, the comparison of the automatic transcription against what was said leaves little room for subjectivity. On the other hand, in the case of TTS, human judgment on naturalness is fairly subjective. Where is AER sitting along that scale? Is a proportion of subjectivity skewing the prediction of user experience from standard classification error rates? While formal studies on that topic remain to be led, a few field observations can be reported:

- False alarms have a negative impact on user opinion if they are too frequent.
- True positives are also annoying if they are too frequent, thus requiring either to build black-out periods into the system, or to report a true positive *after the sound has sounded for a while*, at the expense of system latency.
- Missed detections can have a dramatic impact on product reputation if the application is related to controlling some critical smart home feature, in particular security, but also comfort.
- Experiencing missed detection might be far from obvious and very rare for certain sounds: most people will probably not break their own windows to test if the system works. On the other hand, baby cries or smoke alarms can be more easily accessed; therefore, potential missed detections for these sounds will be more likely to trigger negative user opinions. The point here is that potential missed detections might weigh differently on user experience, depending on the level of ease with which the related sound can be voluntarily triggered by the users themselves.
- False alarms, if they can be checked against the sound snippet which triggered the alarm, do not all have the same level of negative impact on user's opinions:

they can be forgiven in the cases where users are able to form some concept of sound proximity or to imagine a reason for the audio detection. As a thought exercise, a potential confusion between baby cry and female opera singing might trigger less of a negative opinion than a confusion between a baby cry and the sound of a vacuum cleaner, because users may be able to imagine that baby cry and female opera singing are both human, vocal, forceful, high-pitched sounds. The hypothetical confusion between a human-generated sound and a machine noise seems inherently harder to forgive because users hear semantic fields, not the acoustics of sounds per se. In some cases, the distinction might be even more subtle, due to involving more complex semantic connections beyond the characterization of object or process which produced the sound: a hypothetical confusion between a smoke alarm and a beepy phone ringtone may be hard to forgive, in spite of both sounds being defined as alerts generated by similar electronic processes, because the smoke alarm is a sign of danger, whereas the phone ringtone may simply be a welcome sound.

Qualitative Assessment of Errors In relation to the notion of qualitative difference between false alarms, studies in the music information retrieval domain [48, 59] have underlined that quantitative classification accuracy may not be enough to characterize whether the underlying algorithm is delivering the functionality that users expect from it. In particular, there is a possibility for any machine learning system to behave as a “horse,” with reference to the story of “Clever Hans” [60, 76], where the system may be altogether solving a different problem than the one it is designed for, thus delivering different functionality than the one expected by the users once it is deployed in the field. Indeed, horse behavior may not be immediately apparent from the error rates, but it may become apparent from the deeper qualitative assessment made by human listeners of what the system does in general, and in particular the qualitative analysis of the errors made by the system in relation to the desired use case and experience of that use case.

The suggestion here is that standard quantitative error rates, such as the F-scores and equal error rates, are not necessarily good predictors of user experience due to the “subjectivity gap” and the requirement to deliver on use cases rather than error rates. There is therefore a need, at the product design and AER applications level, to introduce metrics which are more directly involving user opinion, and to propose metrics which would bridge the gap between standard error rates and user experience. Access to user experience at the fundamental research level may thus suggest to build a physical exemplar of a “listening object” to bridge the gap between error rates and an actual application. At the simplest level, this can be done with a standard computer, a microphone, and some simple form of alerting system. As an example of how this may be very beneficial to research, a domain where the building of physical instances of the system has considerably helped to steer research directions is that of meeting room transcription [31], where virtually every speech recognition research lab in the late nineties and early 2000s started to build its own transcriber based on standard ASR technology, soon to realize that

far field capture, speaker diarization and the processing of expressive speech were going to become crucial research topics. In this line of idea, encouraging academic researchers to build a sound recognition exemplar in order to surface salient research problems might be a good idea, and can be done either independently or as a partnership with industrial labs.

Subjective Metrics The development of metrics which correlate well with user experience depends on having access to user experience measurements to begin with. Opinion scoring methodology is a solution to that, which has been extensively deployed and studied in the speech synthesis domain [16], and to some extent in music information retrieval [48]. In that framework, two points need to be kept in mind:

- Opinion scores are Likert-type scales and inherently ordinal, so standard arithmetic means cannot be used reliably across such scores [16]. However, boxplots and a careful analysis of significance intervals via the Wilcoxon signed rank test [16] do yield useful ways of comparing systems with opinion scoring.
- In opinion scoring, what is being ranked depends strongly on the question which is being asked to the users [20]. For example, in speech synthesis, it is a different thing to ask:

“Do you like this system: (1) strongly dislike (2) dislike (3) neither like nor dislike (4) like (5) strongly like”

than to ask:

“Please rate the naturalness of the system: (1) very dissimilar to human (2) somewhat dissimilar to human (3) somewhat similar to human (4) similar to human (5) very similar to human”.

In the above example, pleasantness and naturalness may be two different notions, as it is possible to design TTS cartoon voices which are pleasant but unnatural. Similarly, in the case of rating an AER system, it will be a different thing to ask, e.g., about system usefulness, than to ask about system annoyance:

“Was this sound detection alert annoying: (1) very annoying (2) annoying (3) neither annoying nor welcome (4) welcome (5) very welcome”

or about relevance of the alerts:

“Do you understand why the system sent a baby cry alert?”

or

“Did you expect to receive a baby cry alert for this sound?”

The above examples may all yield a different type of insight into the system, but also different numerical ranges for the related opinion scores.

Opinion scoring questions can be included by design in the user interface of smart home products, in order to measure system improvements in real time.

12.3.3 *Distinguishing Objectivity from Subjectivity in AER Evaluation*

Once human-generated opinion scores are available, then it becomes possible to investigate if existing error metrics correlate well with these [44, 48], and also to investigate new error metrics or algorithms which correlate better with user experience. For example, it has been reported in [48] that optimizing a music boundary detection system against F_α scores rather than F_1 scores, where the value of α was emphasizing precision over recall, was leading to a system receiving better opinion scores. The idea here is that an extra parameter, α in this example, may allow the tuning of the correspondence between the “objective” metric and the subjective opinion scores. “Objective” is put between quotes here, because it should be kept in mind that the F-score, precision, and recall measures actually depend on the subjective, application-dependent choice of an operation point by the system designer.

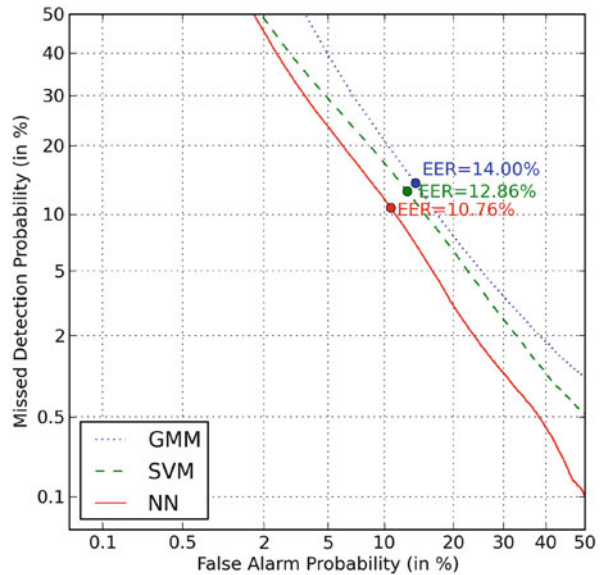
As a matter of fact, most detection systems make a decision by thresholding a score, which results by design in the amount of false alarms (FAs) and missed detections (MDs)⁴ being interdependent: the threshold can be made either more permissive, by letting more events through in general, which results in reducing the false negatives at the expense or increasing the false alarms, or it can be made more conservative, by letting less events through, with the effect of reducing the false positives at the expense of increasing the false negatives. The determination of the threshold and FA/MD compromise according to applicative constraints is known as setting an operation point for the system, also known as performing the calibration of the system, and it is subjective to the application of interest: some users may subjectively prefer a more conservative system to a permissive one if they are excessively annoyed by false alarms, whereas other users may prefer to get as many alerts as possible, because worrying about the possibility of a missed detection outweighs the annoyance related to false alarms.

This consideration affects the definition of what “the best system” means in a comparative evaluation: are we talking about the system which has the best ability to model acoustics and to discriminate between sounds, or the system which fits the use case best? From that perspective, the possibility exists that precision, recall, and F-score may miss the best acoustic model if the operation point was wrongly chosen for the application of interest, or if a diversity of subjective operation point assumptions were made across the compared systems.

This problem of distinction between objective discrimination and subjective calibration in system evaluation is extremely well explained and studied in the context of speaker recognition in [71], with more technical detail in [11] and further developments in [10]. In these studies, solutions to this problem are proposed in the form of:

⁴ False alarm (FA) is synonymous to false positive (FP), and missed detection (MD) is synonymous to false negatives (FN). Usage varies across domains in the literature, e.g., speaker recognition tends to use the former more often.

Fig. 12.2 Example of DET curves used to compare the acoustic modeling power of various systems in [57], independently of the subjective choice of an operation point



- systematically using detection error trade-off (DET) curves [7, 41], to compare sound discrimination globally across the full range of operation points,
- complementing DET curves with discrete cost function (DCF) figures [7, 41, 71], log-likelihood ratio cost function (C_{llr}) measures and applied probability of error (APE) curves [71], all described in more detail below, to achieve extra insight into discrimination versus calibration.

DET curves [7, 41] are warped versions of the receiver operating characteristic (ROC) curves introduced in Chap. 6, where the use of normal-deviate scaling flattens the curve and facilitates the visual comparison of systems. In this type of plot, DET curves globally closer to the origin correspond to better sound discrimination (e.g., Fig. 12.2). DET curves are often summarized into a single equal error rate (EER) figure, which corresponds to the point where the curve crosses the $P_{\text{miss}} = P_{\text{FA}}$ diagonal. It can be demonstrated [10] that EERs allow to rank the discriminative ability of various systems globally and independently of the choice of an operation point. However, DET curves may cross each other, meaning that for some ranges of operation points, the ranking of systems after calibration may end up inverted. In this line of idea, it should be kept in mind that the EER operation point, where the missed detection rate is equal to the false alarm rate, might be irrelevant to open-set 24/7 applications, where the system should intuitively be more guarded against false alarms due to the vast exposure of the system to non-target sounds. For example, looking at Fig. 12.2, it might be the case that the dashed curve would cross the solid curve for conservative operation points where the system would be expected to achieve, e.g., false alarm rates under 1%, thus questioning which system would be “the best” for that section of the curve.

Following this, and in contrast to the EER, the discrete cost function (DCF) seeks to incorporate information about the relative cost of false alarms and missed detections, and is thus defined as:

$$C_{\text{det}}(P_{\text{miss}}, P_{\text{FA}}) = C_{\text{miss}}P_{\text{miss}}P_{\text{tar}} + C_{\text{FA}}P_{\text{FA}}(1 - P_{\text{tar}}) \quad (12.1)$$

The above definition introduces the cost parameters C_{miss} and C_{FA} to subjectively weigh the relative importance of the objective missed detection rate P_{miss} and objective false alarm rate P_{FA} into the evaluation, given the prior probabilities of occurrence of the target sound P_{tar} versus occurrence of non-target sounds $(1 - P_{\text{tar}})$ in the application of interest, and for an operation point $(P_{\text{miss}}, P_{\text{FA}})$ subjectively chosen along the DET curve by setting an application-dependent decision threshold θ . As such, the role of C_{miss} and C_{FA} is similar to that of α in the F_{α} measure, i.e., to tailor the evaluation metric to applicative subjectivity.

However, the DCF is a single number happening after the choice of a decision threshold. The work in [10, 11, 71] therefore seeks to develop a more global measure of performance across the whole range of possible thresholds and possible subjective cost values. This is done by defining the log-likelihood ratio cost function C_{llr} as the integral over all possible decision thresholds and costs of all the possible DCFs. In the course of that, a set of mathematical tricks is used to clarify the separation between the global measurement of classification errors and the parameters related to calibration, by defining the following quantities:

- The total probability of error

$$P_e(\theta) = \tilde{P}_{\text{tar}}(\theta)P_{\text{miss}}(\theta) + (1 - \tilde{P}_{\text{tar}}(\theta))P_{\text{FA}}(\theta) \quad (12.2)$$

bundles the $P_{\text{miss}}(\theta)$ and $P_{\text{FA}}(\theta)$ into a single metric via weighting by the prior log-odds $\tilde{P}_{\text{tar}}(\theta)$, which will be explained below. In this equation, all the quantities in play are functions of the threshold θ . As such, $P_e(\theta)$ expresses the DCF as a function of the threshold rather than as a single number.

- In the above equation, the trick is to realize that the calibration parameters C_{miss} , C_{FA} and θ are redundant against the data set characteristic P_{tar} (please see [71] for more detail), and can thus be grouped into a single definition for the decision threshold θ :

$$\theta = \log\left(\frac{P_{\text{tar}}}{1 - P_{\text{tar}}}\frac{C_{\text{miss}}}{C_{\text{FA}}}\right) \quad (12.3)$$

Thus, the quantity \tilde{P}_{tar} , referred to as prior log-odds, can be defined as

$$\tilde{P}_{\text{tar}} = \frac{P_{\text{tar}}C_{\text{miss}}}{P_{\text{tar}}C_{\text{miss}} + (1 - P_{\text{tar}})C_{\text{FA}}} \quad (12.4)$$

$$= \frac{1}{1 + e^{-\theta}} = \text{logit}^{-1}(\theta) \quad (12.5)$$

where $\text{logit}(x) = \log\left(\frac{x}{1-x}\right)$. With this definition of the threshold, all the calibration parameters are now grouped into θ , or equivalently into the warping function $\tilde{P}_{\text{tar}}(\theta)$. Thus, $P_e(\theta)$ can be plotted versus θ via a warping of the threshold scale, or equivalently a warping of the classification scores issued by the classifier, via the logit warping function. This plot of $P_e(\theta)$ is called the applied probability of error (APE) curve [71].

- From there, the log-likelihood ratio cost function C_{llr} can be defined as the integral of $P_e(\theta)$ across θ , or equivalently the integral of all the possible DCFs across the range of decision thresholds and classification outcomes, i.e., across the whole spectrum of operation points for a given DET curve, and where the information about the calibration is now fully contained within the warping function/prior log-odds $\tilde{P}_{\text{tar}}(\theta)$. This is different from the area under curve (AUC, see Chap. 6), which is the integral of the DET curve (or rather its equivalent ROC curve), and as such does not contain any information about calibration, only about discrimination, and reports a quantity which is more difficult to interpret as an error measure.
- From there, [11, 71] propose a way of minimizing the C_{llr} by optimizing the warping function applied to θ , or in other terms to optimize the $\tilde{P}_{\text{tar}}(\theta)$ function which will non-linearly rescale the probability densities $P_{\text{miss}}(\theta)$ and $P_{\text{tar}}(\theta)$ in order to find the “truly minimal error” P_e globally across the whole θ spectrum, given considerations of both discrimination ($P_{\text{miss}}(\theta)$ and $P_{\text{tar}}(\theta)$ contained into the DET/ROC curve) and calibration (the calibration parameters contained into the optimal warping curve $\tilde{P}_{\text{tar}}(\theta)$).

Along a similar line of thought, i.e., clarifying the separation of discrimination from calibration, the open-set optimization method introduced in [56] makes room for the explicit inclusion of applicative constraints into the optimization process, by allowing to bound the precision and recall rates as one of the constraints playing into the optimization method. As such, this method can be thought of as accounting for user opinion directly into the training process, rather than operating calibration and training independently from each other.

As of now, this general problem of discrimination versus calibration, or in other terms achieving application-independent methodology for system comparison, seems to have remained understudied into AER research. Indeed, we have not been able to find mentions of the usage of C_{llr} and APE curves in AER literature, in spite of the attractive solution that these methods represent to get around some of the pitfalls of comparing systems on the basis of F-score, precision, and recall only. On the short term, our suggestion would be to resort to the computation of the F-score only as a by-product of setting an operation point along the DET curve, and stating explicitly whether the evaluation was made under a subjective assumption of conservatism or permissiveness of the system, or otherwise more explicitly stating the chosen applicative constraints, in order to avoid implying that the ranking of the compared systems qualifies some notion of best acoustic modeling power in a universal and objective way. Or in simpler terms, the proportion of subjective calibration that exists in today’s standard evaluation metrics should probably not be

ignored, while interest for more application-independent metrics should probably be further developed.

12.3.4 Summary: Objectivity, Subjectivity, and User Experience

The research point generally made in this subsection is that AER systems, and machine learning systems in general, are designed to be used by humans, with some applications in mind (for example, smart home applications). The applications themselves trigger constraints on system calibration, i.e., the choice of an operation point which is agreeable to user in terms of the balance between false alarms and missed detections, but also the acoustic and semantic nature of these errors. User opinion can be measured on this, e.g., by using opinion scores. Insofar as standard error metrics such as precision, recall, and F-score inform jointly about the discrimination and the calibration, with possible adjustments towards particular operation points in the case of the F_α or the DCF scores, finding which AER system is “the best” in the general terms of being the most acoustically discriminative across a wide range of applications may currently suffer from the understudied dependency on the application-related subjectivity of system calibration. The hope of this section is to attract attention on this problem, and to trigger an evolution of mindsets where AER methodology would either claim more explicitly the application within which the comparison is being performed, including an explicit assessment of the relative costs of false alarms and missed detection on user opinion, or would adopt evaluation methods which would allow to find “the best system” more independently from calibration and applications.

12.4 Ethical Issues: Privacy and Data Protection

Another aspect of AER applications which affects user opinion is the question of privacy and ethics surrounding a system which artificially “listens” to people’s personal environment. Indeed, the collection of audio data in people’s homes, for the purpose of both AER fundamental research and commercial applications, generally falls under the principles of data protection laws and ethical recommendations. However, legal requirements vary widely between countries. After an overview of the situation in this domain, this section will focus on European law, which is the one introducing the most comprehensive set of data protection rules. European requirements in this field can be summarized and understood through a list of key concepts, which translate into best practice when it comes to gathering audio data in home environments and processing it for AER research and development.

12.4.1 Which Country are We Talking About?

The formal concept of a *right to privacy* has started to emerge in international law following World War II. It is explicitly encoded in Article 12 of the *Universal Declaration of Human Rights* [65], as well as in Article 8 of the *European Convention on Human Rights (ECHR)* [26] which followed 2 years later with a similar but more substantiated formulation. These declarations of right to privacy have entailed the creation of data protection laws in various other countries internationally. Although most of the data protection laws were written with the protection of textual data in mind, the processing of audio data recorded in people's personal space is usually acknowledged to fall within the remit of these laws [68]. However, reference to audio data is not always made explicitly, thus leaving room for interpretation and complicating compliance checks. Furthermore, the requirements of data protection laws are largely region dependent:

- **Europe:** Europe has established strict and elaborate data protection laws and policies, the detail of which can be grasped by reading a number of documents [25, 27, 33], the most recent of which being the *European General Data Protection Regulation 2016/679* [63] which will take effect in May 2018 and will unify and substitute for each Member State's national laws in the domain (e.g., [40, 67]). Each country usually implements a government body in charge of explaining and controlling compliance to data protection laws, such as, e.g., the Commission Nationale de l'Informatique et des Libertés [17] in France or the Information Commissioner's Office [34] in the UK. It can be useful for AER researchers to consult with these resources in order to get advice and checks on the legal and ethical compliance of their research and data handling policies.
- **USA:** US data and privacy protection laws [8, 75] are very heterogeneous as they are sector specific (e.g., different laws governing the collection and disclosure of financial data, health-related data, etc.) as well as state specific (i.e., 13 US states out of 50 have enacted a variety of digital privacy and data protection laws over the years), yet contained within specific Federal laws which empower US Federal agencies to process personal data. Although US citizens may invoke protection through the Fourth Amendment and the Privacy Act, US data protection laws are generally described as far less comprehensive than the EU data protection framework [8, 75]. Because US law can be fairly complex, legal compliance is usually facilitated by institutional review boards (IRBs) [78], which are committees local to various research institutions (e.g., universities) and are empowered to approve, require to modify, or disapprove research plans, from the standpoint of preserving the rights and welfare of human research subjects. As such, local IRBs would be the main point of contact to check the legal and ethical compliance of AER research led in the USA.
- **Asia and rest of the World :** The situation varies widely across Asia and the rest of the world. On the one hand, a number of countries have taken European Directive 95/46/EC as a template, e.g., Japan [37] and South Korea [36]. In contrast, other countries have implemented only minimal data protection law

through sector specific acts, mostly to answer demands from foreign investors. For example, China [22], Vietnam [14], and many other countries [49] do not have a data protection regime according to EU's definitions, and their approaches to personal data processing and the right to privacy remain mostly aimed at the individual as a consumer.

This worldwide diversity suggests that the ethical notions surrounding the processing of audio data, and related requirements of privacy protection, may depend on the level of economic development, but also equally importantly on cultural factors, e.g., if realizing that the ancestry of European data protection laws is directly traceable back to the traumas of World War II. From a practical perspective, the processing of personal audio data may be less regulated and thus perceived as easier to manage in certain countries than others, although it may be perceived as more or less ethical according to various cultural standard. Finally, compliance checks and information sources about ethical research in AER depend on a variety of country-dependent institutions, which are useful sources of information on ethics and legal compliance.

Going forward, this section will focus on the most comprehensive and strictest privacy and data protection scenario, which is the European data protection regime.

12.4.2 Is Environmental Audio Data Actually Private?

When it comes to fulfilling the legal and ethical responsibilities surrounding the manipulation of audio data, two points are important to understand: what are the legal rights attached to personal data, and in which cases audio data falls under the definition of private data, thus triggering special attention to legal compliance.

What can and cannot be done with personal data in Europe is contained within the definition of the rights given to individuals by the European data protection acts (DPAs) [25, 63, 64, 69]⁵:

- a right of access to a copy of the information comprised in their personal data;
- a right to object to processing that is likely to cause or is causing damage or distress;
- a right to prevent processing for direct marketing;
- a right to object to decisions being taken by automated means;
- a right in certain circumstances to have inaccurate personal data rectified, blocked, erased, or destroyed; and
- a right to claim compensation for damages caused by a breach of the Act.

⁵Many of the definitions in this section are quoted from the ICO's documentation [69, 70] and the UK Data Service's website [64]. Both services are local to the UK and government funded, and they aim at helping researchers and businesses understanding the legal requirements set forth into the UK's Data Protection Act 1998. Because the UK's DPA 1998 in itself seeks to implement European recommendations and directives, the notions quoted in this list are echoing the general definitions set forth in the various European laws and recommendations.

The responsibility to comply with the law on that relies on the *data controller* [70], who is legally responsible for determining the purposes for which and the manner in which any personal data are collected and processed. A lab doing research on AER may or may not be the data controller, as there is another legally distinct notion of *data processor*, which means any person, other than an employee of the data controller, who processes the data on behalf of the data controller, where processing includes pretty much anything that can be done for data collection and AER research and development: recording, altering, listening to, aligning, transferring, deleting, etc. If there is evidence of a breach of the DPA, it is the data controller involved that could be subjected to prosecution.

It should be understood that the above-mentioned rights aim specifically at the protection of various levels of *personal data*, defined as follows [64, 69]:

- **Personal data** are data which relate to a living individual who can be identified from those data or from those data and other information which is or is likely to come into the possession of the data controller (for example, a name and address) and includes any expression of opinion about the individual and any indication of the intentions of the data controller (for example, selling something to persons of a certain age). This includes any other person in respect of the individual, for example, someone's parents or children. It is allowed to store and process personal data, as long as it is not sensitive data, confidential data or does not lead to profiling.
- **Profiling** is defined as the task of inferring sensitive data from derived data capture.
- **Sensitive data** is defined as data that could be used for unlawful discrimination, for example, data indicative of ethnic origin, political opinions, religious beliefs, sexual orientations etc. according to the law's definitions. The audio recordings must not target such data, and any data that would be found in the recordings and which could be qualified as sensitive according to the law's definitions should be immediately and permanently deleted.
- **Confidential data** are data given in confidence or agreed to be kept confidential, i.e., secret, between two parties, that are not in the public domain such as information on business, income, health, medical details, and political opinion. There again, any confidential data captured in audio recordings should be deleted.

When it comes to environmental audio recordings, the question of compliance with local DPAs therefore boils down to assessing whether any of the above data categories are being processed by the AER system, and for what purpose.

As soon as the recordings involve speech, it is more difficult to ensure that the data does not contain any name, phone number or otherwise identifying elements, and it can be argued that voice contains identity information of a biometric nature. Therefore, it is safer to treat speech recordings as personal data, unless it can be explicitly proven that the recordings cannot be traced back to a particular individual in any way (e.g., in the case of incidental recordings in public places). Recordings made in a home have a higher probability of containing identifying data

and a relation to the home address, and should thus be treated as personal data for the purpose of the local DPA.

When it comes to other types of sounds than speech, as of now it can be argued that AER algorithms are unable to infer sensitive information or to provide profiling according to the law's definition. As a matter of fact, the direct goal of AER systems is to label general audio events and audio scenes for practical events such as, e.g., glass breaking or babies crying, which by themselves are not semantically rich enough to deliver profiling or to help inferring sensitive data according to the law's definition. However, cases where sound recognition could allow profiling when aggregated with other sensing modalities remain open to the imagination. Therefore, it still belongs to the researcher, product manager or data controller in general to carefully consider and justify the absence of such misuses of AER technology. Indeed, most European funding application processes will ask for clear statements on that, and companies as much as academic institutions will want to prepare answers to avoid legal exposure.

It should also be noted that although the legal definition of sensitive data and the associated rights are narrowly defined by the law and exclude a very vast majority of what may be recorded or processed for AER purposes, legal and ethical guarantees may not actually bring a full and complete answer to the public's concerns related to a broader and more intuitive sense of privacy or feelings about being eavesdropped, especially if the audio data capture happens in a space as personal as home. Additional notions, such as consent and data protection, are thus also feeding into the management of the public's perception of privacy.

12.4.3 Consent and Ownership

From a legal and ethical perspective, whatever will be done with personal data must be done with the recorded person's consent, and must not consist in profiling or discrimination that would be operated without that person's knowledge and consent [18]. Consent to data usage must be unambiguous, informed, freely given, specific, and explicit. In practice, this translates into designing a set of documents and procedures to inform the users about the purpose of the data collection and their legal rights in this matter:

- **Data usage information:** In this document, the purpose of the data collection must be clearly and credibly stated to be solely that of AER research, improving sound recognition algorithms and the development and improvement of sound recognition algorithms or products. The redaction of this document is the right place to analyze the risks of the technology allowing potential usage for profiling, and avoiding such risks.
- **Data sign-off procedure:** a process of audio data reviewing and sign-off can be put in place to allow the recorded users to review the recorded data and address any privacy concern before finalizing the data control and ownership transfer.

- **Data protection policy:** this document should describe the measures put in place by the data controller to protect the data, for example, secure storage accessible only by the data controller and the data processors. It should also give details about the rights of the recorded subjects to access a copy of their data, as well as details on how to request deletion of the data if they can legally prove that its processing is causing them damage or distress a posteriori from the transfer of data ownership.
- **Transfer of data control and ownership** from the recorded subjects to the research labs and/or the company involved in the audio data collection must be legally framed by a contract between both parties. It is usual to mention a fee in the contract to materialize consideration for the data exchange, even if the recorded volunteers were keen to give their audio data for free. The fee can range from a symbolic fee (e.g., symbolic Euro or symbolic British Pound), or any other reward offered against the collected data (e.g., shopping vouchers when dealing directly with volunteers), up to substantial licensing fees when transferring the control of a fully post-processed data set between corporate and/or academic entities.

It is important to keep in mind that *all* inhabitants of a household, or the legally responsible person in the case of children, should give their consent to audio data collection for the usage of this data to be lawful and ethical. Thus, households whose inhabitants are all unable to give their informed consent should better be avoided. Households involving inhabitants both able to give their informed consent and unable to do so (e.g., families with children or handicapped relatives, who could strongly benefit from the developed technology) should be considered and handled with specific information and consent forms. In most cases, it is advisable to contact the legal department of one's academic institution or a private law firm to check the legality of the information sheets and ownership transfer contracts, prior to starting an AER data collection project.

Apart from the awareness of such legalities, technical solutions can be sought to put the users themselves in control of their privacy. For example, audio data capturing devices can be equipped with an "off" or "listening mute" button that can be freely used by the end users whenever they would like audio to remain excluded from the data collection or processing. Automating this by predicting or preempting the desire for privacy is an open research question in itself, with, e.g., the automatic detection of whether the users are present in the room or not, or the automatic preemption of privacy levels when the end users are still present in the audio scene (e.g., at the simplest level, speech detection raising the privacy level). Privacy preserving algorithms as discussed in the next section may also address this issue.

12.4.4 Data Protection and Security

The concept of *machine listening* immediately raises intuitive concerns about the risk of eavesdropping [19], which is a general concern for any Internet of Things application where sensor data gets streamed to the cloud [15].

At the surface level, appropriate data security and confidentiality measures must be put in place to protect audio data transmission and storage, for example, access control and encryption [15], thus ensuring that audio processing applications cannot be breached into and cannot be used for eavesdropping. In the case where a lab is a data processor rather than a data controller, the level of security must be as good as, or exceed, the data controller's own standards, which can be a pitfall if working with industrial partners whose security standards are high and costly.

However, access control measures may not be sufficient to reassure the general public, because they address the access to the contents but do not address the scrambling of the contents itself, under the belief that access could be breached. There is a dilemma for AER research and development there, as scrambled data cannot be labeled and is thus of limited usefulness for debugging and system training.

Thus, at the technical level, data protection and security may also trigger interest into the following areas:

- **Processing on the edge:** realizing AER directly on embedded devices inside people's home and avoiding processing in the cloud guarantees that private audio data does not leave the user's home in any way, thus delivering a stronger guarantee of privacy protection. But, this requires the deployed algorithms to run at a sufficiently low computational cost to fit on embedded devices, as discussed in Sect. 12.2.3. Privacy requirements may thus have the effect of indirectly imposing a limit on sound recognition performance, if the infinite computational power available in cloud computing is not an acceptable option.
- **Investigating anonymization techniques** [18]: similarly to vision, where object detection solutions can be imagined to de-personalize the content (e.g., face detection followed by blurring), investigating de-personalization techniques could be done for AER. At the simplest level, the detection and deletion of speech could be one way forward. However, for particular AER tasks such as automatic scene classification or event detection related to human activities such as, e.g., aggression detection, removing speech runs the risk of destroying the coherence of the audio scene entirely.
- **Privacy preserving algorithms:** Privacy preserving speech processing [51, 52] has recently emerged as a full-fledged research topic in this area, and could possibly be extended to more generic audio scenes.

So far, AER research seems to have focused more on sound recognition performance than on such privacy aspects, which were more traditionally deferred to the industrial development stages. However, the above points suggest that research on AER algorithms can play a key role in finding solutions to address the public's concerns about audio data privacy and protection.

12.5 Conclusion

After giving a brief overview of the relevance and value of deploying automatic audio event recognition (AER) in the smart home market, this chapter has reviewed three aspects of the productization of AER which are important to consider when developing pathways to impact between fundamental research and “real-world” applicative outlets:

- In Sect. 12.2, it is shown that applications introduce practical constraints on the productization of AER algorithms. One constraint is to achieve a precise definition of what a sound event should be, a point which relates to evaluation of the system but also to acoustic modeling. Proposals are made to move away from the bag of frames approach, in order to focus more on the investigation of temporal modeling as well as the explicit modeling of interruption. Another constraint is to run and evaluate AER for 24/7 constantly listening applications, where it becomes impractical to enumerate the set of non-target sounds. The proposal there is to recast the problem as one of open-set recognition, with pointers to preliminary work on that in the image recognition domain, as well as a suggestion of evaluating false alarm rates as a function of time rather than using absolute event counts. The final constraint is that of running AER applications on consumer devices, which have imperfect microphones and limited computational capabilities. In this context, new research should focus on directions which may not have been extensively explored as of yet, e.g., robustness against channel and room effects, and scalability or performance as function of the computational cost.
- Section 12.3 explores the definition of user experience for AER, a notion which it is crucial to optimize in order to achieve some usefulness of AER for practical applications. After reporting field observations about the way various errors may affect user experience and user opinion in various ways, it is proposed to introduce opinion scoring in AER evaluation methodology. Then, the question of whether standard AER performance metrics reflect the quality of user experience is being explored, and attention is being drawn to the fact that standard metrics mash up a proportion of objective evaluation of the system’s ability to discriminate between sounds, with a proportion of subjectivity related to the choice of an application-dependent operation point. Solutions to the separation of discrimination and calibration in system evaluation are introduced, and inspired from the speaker recognition domain. Generally, the point there is to distinguish more explicitly between two separate definitions of “best system,” one which focuses on finding the model which offers the best acoustic modeling power in general, versus another one which focuses on optimizing user experience.
- Finally, in Sect. 12.4, the deployment of AER in the field is analyzed from the standpoint of introducing the ethical and legal requirement to respect the users’ fundamental right to privacy, in particular with regard to systems which are “listening” at all times into the users’ private spaces. A review of the key notions underpinning European laws in the domain of data and privacy protection, which

are the most comprehensive across the world, suggests that lawful and ethical use of audio data amounts to empowering users to consent by fully informing them about the use of their data, as well as taking reasonable information security measures to protect the users' personal data.

Insight into these three topics is being contributed with the hope of shortening the path that leads from fundamental AER research to the social and economical impact entailed by deploying AER in the field of smart home applications.

Acknowledgements The research work exposed in Sect. 12.2.3 has been developed as a collaboration between Queen Mary University of London and Audio Analytic, supported by InnovateUK grant nr.131604 and EPSRC grants EP/M507088/1 & EP/N014111/1, as well as private funding from Audio Analytic Ltd. Prof. Mark Plumbley, currently at University of Surrey, has supervised the work of Dr. Siddarth Sigtia, currently with Apple, and Dr. Adam Stark, currently with Mi.mu Gloves, on the academic side of this research work. Tamara Sword from Audio Analytic has contributed some of the wording about use cases and marketing aspects. The author would like to thank Dr. Tuomas Virtanen and Dr. Juan Bello for their insightful comments on this chapter.

References

1. Ahuja, K., Schneider, J., de Maisieres, M.T.: *The Connected Home Market*. McKinsey & Company, New York (2015). http://www.mckinsey.com/spContent/connected_homes/pdf/McKinsey_Connectedhome.pdf
2. Aldrich, F.K.: Smart homes: past, present, future. In: Harper, R. (ed.) *Inside the Smart Home*, pp. 17–39. Springer, London (2013)
3. Aucouturier, J.J., Defréville, B., Pachet, F.: The bag-of-frames approach to audio pattern recognition: a sufficient model for urban soundscapes but not for polyphonic music. *J. Acoust. Soc. Am.* **122**(2), 881–891 (2007)
4. Battaglino, D., Lepauloux, L., Evans, N.: The open-set problem in acoustic scene classification. In: 2016 IEEE International Workshop on Acoustic Signal Enhancement (IWAENC), pp. 1–5 (2016)
5. Bendale, A., Boulton, T.E.: Towards open set deep networks. CoRR abs/1511.06233 (2015). <http://arxiv.org/abs/1511.06233>
6. Biet, N.: *Internet of Things - Overview of the Market*. The Factory, Belgium (2014). <http://www.thefactory.com/wp-content/uploads/2015/01/IoT-market-overview-Final.pdf>
7. Bimbot, F., Bonastre, J.F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., Reynolds, D.A.: A tutorial on text-independent speaker verification. *EURASIP J. Appl. Signal Process.* **2004**, 430–451 (2004)
8. Boehm, F.: A comparison between US and EU data protection legislation for law enforcement purposes. Technical report, European Parliament (2015). <http://www.europarl.europa.eu/studies>
9. Bonomi, F., Milito, R., Natarajan, P., Zhu, J.: Fog computing: a platform for internet of things and analytics. In: Bessis, N., Dobre, C. (eds.) *Big Data and Internet of Things: A Roadmap for Smart Environments*, pp. 169–186. Springer, Berlin (2014)
10. Brümmer, N.: *Measuring, refining and calibrating speaker and language information extracted from speech*. Ph.D. thesis, Stellenbosch University (2010)
11. Brümmer, N., du Preez, J.: Application-independent evaluation of speaker detection. *Comput. Speech Lang.* **20**, 230–275 (2006)

12. Buchholz, S., Latorre, J.: Crowdsourcing preference tests, and how to detect cheating. In: Proceedings of Interspeech 2011, pp. 3053–3056 (2011)
13. Burges, C.J.: A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Disc.* **2**(2), 121–167 (1998)
14. Business Software Alliance: Global cloud computing scorecard - country report: Vietnam (2012). http://cloudscorecard.bsa.org/2012/assets/PDFs/country_reports/Country_Report_Vietnam.pdf
15. Celesti, A., Fazio, M., Villari, M.: Enabling secure XMPP communications in federated IoT clouds through XEP 0027 and SAML/SASL SSO. *Sensors* **17**, 301 (2017)
16. Clark, R.A.J., Podsiadlo, M., Fraser, M., Mayo, C., King, S.: Statistical analysis of the blizzard challenge 2007 listening test results. In: Proceedings of the Blizzard Challenge (2007). <http://www.festvox.org/blizzard/bc2007/>
17. Commission Nationale de l'Informatique et des Libertés. <https://www.cnil.fr/> (2017). Last accessed 01/2017
18. Corti, L., Van den Eynden, V., Bishop, L., Woollard, M.: *Managing and Sharing Research Data*. Sage Publishing, Thousand Oaks (2014)
19. Crossley, D.: Samsung's listening tv is proof that tech has outpaced our rights. *The Guardian* (2015). <https://www.theguardian.com/media-network/2015/feb/13/samsungs-listening-tv-tech-rights>
20. Dall, R., Yamagishi, J., King, S.: Rating naturalness in speech synthesis: the effect of style and expectation. In: Proceedings of the Speech Prosody Workshop (2014)
21. Davies, M.: C-Sense - exploiting low dimensional models in sensing, computation and signal processing. http://cordis.europa.eu/project/rcn/204493_en.html (2016). European Research Council project ID 694888, hosted at the University of Edinburgh. Online description last accessed 05/2017
22. de Hert, P., Papakonstantinou, V.: The data protection regime in China. Technical report, European Parliament (2015). <http://www.europarl.europa.eu/studies>
23. Deligne, S., Bimbot, F.: Inference of variable-length linguistic and acoustic units by multigrams. *Speech Commun.* **23**, 223–241 (1997)
24. Eskénazi, M., Levow, G.A., Meng, H., Parent, G., Suendermann, D.: *Crowdsourcing for Speech Processing*. Wiley, Chichester (2013)
25. European Council: Directive 95/46/EC of the European Parliament and of the Council (1995). <http://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:31995L0046>
26. European Court of Human Rights, Council of Europe: European Convention on Human Rights (1950). http://www.echr.coe.int/Documents/Convention_ENG.pdf
27. European Parliament: Resolution of 6 July 2011 on a comprehensive approach on personal data protection in the European Union (2011/2025(INI)) (2011). <http://www.europarl.europa.eu/sides/getDoc.do?type=TA&reference=P7-TA-2011-0323&language=EN&ring=A7-2011-0244>
28. Foster, P., Sigtia, S., Krstulovic, S., Barker, J., Plumbley, M.D.: CHiME-Home: a dataset for sound source recognition in a domestic environment. In: Proceedings of the 11th Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (2015)
29. Fulford, N., Sutherland, T.: 'One voice to bind them all' - Smart home devices, AI, children and the law. *Digit. Bus. Lawyer* **18**(10), 12–15 (2016)
30. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (IoT): a vision, architectural elements, and future directions. *Futur. Gener. Comput. Syst.* **29**(7), 1645–1660 (2013)
31. Hain, T., Garner, P.N.: Speech recognition. In: Renals, S., Boulard, H., Carletta, J., Popescu-Belis, A. (eds.) *Multimodal Signal Processing: Human Interactions in Meetings*. Cambridge University Press, Cambridge (2012)
32. Hockett, C.F.: The origin of speech. *Sci. Am.* **203**, 88–96 (1960)

33. Hustinx, P.: EU data protection law: The review of directive 95/46/EC and the proposed general data protection regulation. Technical report, European University Institute's Academy of European Law (2013). https://secure.edps.europa.eu/EDPSWEB/webdav/site/mySite/shared/Documents/EDPS/Publications/Speeches/2014/14-09-15_Article_EUI_EN.pdf
34. Information Commissioner's Office. <https://ico.org.uk/> (2017). Last accessed 01/2017
35. International Organization for Standardization: ISO 8201: Acoustics – Audible emergency evacuation signal. International Organization for Standardization, Geneva (1987)
36. Kim, J.H., Chung, B.T.H., Keh, J.S., Lee, I.H., Kim, I.H., Chang, I.H.: Data protection in south korea: overview. In: Data Protection Multi-Jurisdictional Guide 2015/16. Thomson Reuters, New York (2015). <http://global.practicallaw.com/2-579-7926>
37. Kinoshita, M., Asayama, S., Kosinski, E.: Data protection in japan: overview. In: Data Protection Multi-Jurisdictional Guide 2014/15. Thomson Reuters, New York (2014). <http://global.practicallaw.com/5-520-1289>
38. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
39. Li, R.Y.M., Li, H.C.Y., Mak, C.K., Tang, T.B.: Sustainable smart home and home automation: big data analytics approach. *Int. J. Smart Home* **10**(8), 177–198 (2016)
40. Loi numéro 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés, et convention 108 (1978). <http://www.cnil.fr/>
41. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M.: The DET curve in assessment of detection task performance. In: Proceedings of Eurospeech'97, pp. 1895–1898 (1997)
42. Medaglia, C.M., Serbanati, A.: An overview of privacy and security issues in the internet of things. In: Giusto, D., Iera, A., Morabito, G., Atzori, L. (eds.) *The Internet of Things*, pp. 389–395. Springer, Berlin (2010)
43. Mesaros, A., Heittola, T., Virtanen, T.: Metrics for polyphonic sound event detection. *Appl. Sci.* **6**(6), 162 (2016)
44. Möller, S., Falk, T.H.: Quality prediction for synthesized speech: comparison of approaches. In: Proceedings of the International Conference on Acoustics, pp. 1168–1171 (2009)
45. Möller, S., Kim, D., Malfait, L.: Estimating the quality of synthesized and natural speech transmitted through telephone networks using single-ended prediction models. *Acta Acust. United Acust.* **94**, 21–31 (2008)
46. Murphy, K.P.: Hidden semi-Markov models (HSMMs). Technical report, Massachusetts Institute of Technology (2002). <http://www.cs.ubc.ca/~murphyk/papers/segment.pdf>
47. Nesta, F., Koldovský, Z.: Supervised independent vector analysis through pilot dependent components. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 536–540 (2017)
48. Nieto, O., Farbood, M.M., Jehan, T., Bello, J.P.: Perceptual analysis of the F-measure for evaluating section boundaries in music. In: proceedings of the 15th International Society of Music Information Retrieval Conference (ISMIR) (2014)
49. Norton Rose Fulbright: Global data privacy directory (2014). <http://www.nortonrosefulbright.com/files/global-data-privacy-directory-52687.pdf>
50. Ostendorf, M.: Moving beyond the 'beads-on-a-string' model of speech. In: Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 79–84 (1999)
51. Pathak, M.A.: Privacy preserving machine learning for speech processing. Ph.D. thesis, Language Technologies Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA (2012)
52. Pathak, M.A., Raj, B., Rane, S., Smaragdus, P.: Privacy-preserving speech processing: cryptographic and string-matching frameworks show promise. *IEEE Signal Process. Mag.* **30**(2), 62–74 (2013)
53. Pragnell, M., Spence, L., Moore, R.: *The Market Potential for Smart Homes*. Joseph Rowntree Foundation, York (2000)
54. Renals, S., Bourlard, H., Carletta, J., Popescu-Belis, A.: *Speech Recognition*. Cambridge University Press, Cambridge (2012)

55. Reynolds, D.: Gaussian mixture models. In: Li, S.Z., Jain, A. (eds.) *Encyclopedia of Biometrics*, pp. 659–663. Springer, Berlin (2009)
56. Scheirer, W.J., Rocha, A., Sapkota, A., Boulton, T.E.: Towards open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell. (T-PAMI)* **36**, 1757–1772 (2013)
57. Sigita, S., Stark, A.M., Krstulović, S., Plumbley, M.D.: Automatic environmental sound recognition: performance versus computational cost. *IEEE/ACM Trans. Audio Speech Lang. Process.* **24**(11), 2096–2107 (2016)
58. Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.D.: Detection and classification of audio scenes and events. *IEEE Trans. Multimedia* **17**(10), 1733–1746 (2015)
59. Sturm, B.L.: Classification accuracy is not enough. *J. Intell. Inf. Syst.* **41**(3), 371–406 (2013). <http://rdcu.be/m8F6>
60. Sturm, B.L.: A simple method to determine if a music information retrieval system is a “horse”. *IEEE Trans. Multimedia* **16**(6), 1636–1644 (2014)
61. Su, L., Yeh, C.C.M., Liu, J.Y., Wang, J.C., Yang, Y.H.: A systematic evaluation of the bag-of-frames representation for music information retrieval. *IEEE Trans. Multimedia* **16**(5), 1188–1200 (2014)
62. Temko, A., Malkin, R., Zieger, C., Macho, D., Nadeu, C., Omologo, M.: CLEAR evaluation of acoustic event detection and classification systems. In: Stiefelhof, R., Garofolo, J. (eds.) *Multimodal Technologies for Perception of Humans*, pp. 311–322. Springer, Berlin (2006)
63. The European Parliament and the Council of the European Union: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (2016). <http://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX%3A32016R0679>
64. The UK Data Service: Obligations when sharing data. <https://www.ukdataservice.ac.uk/manage-data/legal-ethical/obligations/> (2017). Last accessed 01/2017
65. The Universal Declaration of Human Rights. United Nations General Assembly resolution 217 A (1948). <http://www.un.org/en/universal-declaration-human-rights/>
66. Tokuda, K., Masuko, T., Miyazaki, N., Kobayashi, T.: Multi-ppace probability distribution HMM. *IEICE Trans. Inf. Syst.* **E85-D**(3), 455–464 (2002)
67. UK data protection act 1998 (1998). <http://www.legislation.gov.uk/ukpga/1998/29/contents>
68. UK Information Commissioner’s Office: Determining what information is ‘data’ for the purposes of the DPA (2012). https://ico.org.uk/media/for-organisations/documents/1609/what_is_data_for_the_purposes_of_the_dpa.pdf
69. UK Information Commissioner’s Office: Determining what is personal data (2012). <https://ico.org.uk/media/1554/determining-what-is-personal-data.pdf>
70. UK Information Commissioner’s Office: Data controllers and data processors: what the difference is and what the governance implications are (2014). <https://ico.org.uk/media/1546/data-controllers-and-data-processors-dp-guidance.pdf>
71. van Leeuwen, D.A., Brümmer, N.: An introduction to application-independent evaluation of speaker recognition systems. In: Müller, C. (ed.) *Speaker Classification I: Fundamentals, Features, and Methods*, pp. 330–353. Springer, Berlin, Heidelberg (2007)
72. Vermesan, O., Firess, P., Guillemain, P., Sundamaeker, H., Eisenhauer, M., Moessner, K., Arndt, M., Spirito, M., Medagliani, P., Giaffreda, R., Gusmeroli, S., Ladid, L., Serrano, M., Hauswirth, M., Baldini, G.: Internet of things strategic research and innovation agenda. In: Vermesan, O., Friess, P. (eds.) *Internet of Things - From Research and Innovation to Market Deployment*, chap. 3. Rivers Publishers, Gistrup (2014)
73. Virtanen, T., Mesaros, A., Heittola, T., Plumbley, M.D., Foster, P., Benetos, E., Lagrange, M. (eds.): *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)* (2016). <http://www.cs.tut.fi/sgn/arg/dcase2016/workshop-proceedings>
74. Wang, Y., Getreuer, P., Hughes, T., Lyon, R.F., Saurous, R.A.: Trainable frontend for robust and far-field keyword spotting. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017)

75. Weiss, M.A., Archick, K.: US-EU Data Privacy: From Safe Harbor to Privacy Shield. Technical report, Congressional Research Service (2016). <https://www.fas.org/sgp/crs/misc/R44257.pdf>
76. Wikipedia entry for “Clever Hans”. https://en.wikipedia.org/wiki/Clever_Hans (2017). Last accessed 01/2017
77. Wikipedia entry for “Displacement”. [https://en.wikipedia.org/wiki/Displacement_\(linguistics\)](https://en.wikipedia.org/wiki/Displacement_(linguistics)) (2017). Last accessed 01/2017
78. Wikipedia entry for “Institutional Review Board”. https://en.wikipedia.org/wiki/Institutional_review_board (2017). Last accessed 01/2017
79. Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T.: Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis. In: Proceedings of Eurospeech’99 (1999)
80. Zen, H., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T.: Hidden semi-markov model based speech synthesis. In: Proceedings of the International Conference on Spoken Language Processing (ICSLP) (2004)
81. Zen, H., Masuko, T., Tokuda, K., Yoshimura, T., Kobayashi, T., Kitamura, T.: State duration modeling for HMM-based speech synthesis. IEICE Trans. Inf. Syst. **E90-D**(3), 692–693 (2007)

Chapter 13

Sound Analysis in Smart Cities

Juan Pablo Bello, Charlie Mydlarz, and Justin Salamon

Abstract This chapter introduces the concept of smart cities and discusses the importance of sound as a source of information about urban life. It describes a wide range of applications for the computational analysis of urban sounds and focuses on two high-impact areas, audio surveillance, and noise pollution monitoring, which sit at the intersection of dense sensor networks and machine listening. For sensor networks we focus on the pros and cons of mobile versus static sensing strategies, and the description of a low-cost solution to acoustic sensing that supports distributed machine listening. For sound event detection and classification we focus on the challenges presented by this task, solutions including feature design and learning strategies, and how a combination of convolutional networks and data augmentation result in the current state of the art. We close with a discussion about the potential and challenges of mobile sensing, the limitations imposed by the data currently available for research, and a few areas for future exploration.

Keywords Urban sound • Smart cities • Noise monitoring • Sensor network • Acoustic sensing • Internet of things (IOT) • MEMS microphone • Audio surveillance • Sound event detection • Sound classification • Machine listening • Machine learning • Deep learning • Convolutional neural networks • Data augmentation

J.P. Bello (✉)
Music and Audio Research Laboratory, New York University,
Suite 1077 35 W, 4th Street, New York, NY 10012, USA
e-mail: jpbello@nyu.edu

C. Mydlarz • J. Salamon
Center for Urban Science and Progress & Music and Audio Research Laboratory,
New York University, New York, NY, USA
e-mail: cmydlarz@nyu.edu; justin.salamon@nyu.edu

13.1 Introduction

13.1.1 *Smart Cities*

Current estimates put the share of the world population living in urban environments at 50%, a number that is expected to grow to as much as 80% by 2050. In OECD member countries, for example, including most of Europe and North America, already 80% of the population lives in cities, with China seeing a net increase of 40% in its share of urban inhabitants during the last 50 years.¹ This rapid trend of urbanization creates massive opportunities for economic development, job diversification, and innovation, but also creates significant problems related to the environmental impact of human activity, the stress to systems and infrastructure, the difficulty of effectively policing and securing public spaces, and potential reductions in health and quality of living for city dwellers.

Unsurprisingly, there is a well-established and growing trend of leveraging technological systems and solutions towards addressing some of the most pressing issues facing urban communities. These *smart cities* initiatives benefit from recent advances in ubiquitous and intelligent sensing, widespread connectivity, and data science to collect, distribute and analyze the data needed to understand the situation on the ground, anticipate future behavior and drive effective action.

13.1.2 *Urban Sound Sensing and Analysis*

The term urban soundscape refers to the sound scenes and sound events commonly perceived in cities. While the specific characteristics of urban soundscapes vary between cities and even neighborhoods, they still share certain qualities that set them apart from other soundscapes. Perhaps most importantly, while rural soundscapes primarily contain geophony (naturally occurring non-biological sounds, such as the sound of wind or rain) and biophony (naturally occurring biological sounds, i.e., non-human animal sounds), urban soundscapes are dominated by anthrophony (sounds produced by humans), which consists not only of the human voice, but of all sounds generated by human-made artifacts including the sounds emitted by traffic, construction, signals, machines, musical instruments, and so on.

Sound is an important source of information about urban life, with great potential for smart city applications. The increase in smart phone penetration and the growing development of specialized acoustic sensor networks mean that urban sound monitoring is becoming an increasingly appealing alternative, or complement, to video cameras and other forms of environmental sensing. Microphones are generally smaller and less expensive than cameras and are robust to environmental conditions

¹<http://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS>.

such as fog, pollution, rain, and daily changes in light conditions that negatively affect visibility. They are also less susceptible to occlusion and are capable of omnidirectional sensing.

The automatic capture, analysis, and characterization of urban soundscapes can facilitate a wide range of novel applications including noise pollution mitigation, context-aware computing, and surveillance. The automatic analysis of urban soundscapes is also a first step towards studying their influence on and/or interaction with other quantifiable aspects of city-life, including public health, real estate, crime, and education.

However, there are also important challenges in urban sound monitoring. Urban environments are among the most acoustically rich sonic environments we could study—the number of possible sounds is unlimited and densely mixed. Furthermore, the production mechanisms and resulting acoustic characteristics of urban sounds are highly heterogeneous, ranging from impulse-like sounds such as gun shots to droning motors that run non-stop, from noise-like sources like air-conditioning units to harmonic sounds like voice. They include human, animal, natural, mechanical, and electric sources, spanning the entire spectrum of frequencies and temporal dynamics.

Furthermore, the complex interaction between this multiplicity of sources and the built environment, which is often dense, intricate and highly reflective, creates varying levels of “rumble” in the background. Therefore, it is not unusual for the sources of interest to overlap with other sounds and to present low signal-to-noise ratios (SNR) that change intermittently over time, tremendously complicating the analysis and understanding of these acoustic scenes.

Importantly, while some audio analysis tasks have a relatively clear delineation between what should be considered a “source of interest” and what should be considered “background” or “noise” (e.g., specific instruments versus accompaniment in music, or individual speakers against the background in speech), this distinction is far less clear in the case of urban soundscapes. Almost any sound source can be a source of interest, and many “noise-like” sources such as idling engines or HVAC units can have similar acoustic properties even though their type and function are very different.

Finally, urban soundscapes are not composed following top-down rules or hierarchical structures that can be exploited as in the case of speech and most music. However, natural patterns of activity resulting from our circadian, weekly, monthly, and yearly rhythms and cultural cycles abound.

13.1.3 Overview of this Chapter

The rest of this chapter is organized as follows. Section 13.2 will briefly discuss the range of applications of automatic sound event analysis and dense sensor networks in urban environments, with a focus on audio surveillance and noise pollution monitoring. Section 13.3 discusses existing solutions for large-scale urban acoustic

sensing and presents the design of a low-cost, scalable, and accurate acoustic sensor network. Section 13.4 provides an in-depth view of the problem of urban sound source identification, and the lessons learned from research efforts to date. Finally, Sect. 13.5 provides a summary of the chapter and some perspectives on future work in this field.

13.2 Smart City Applications

The intelligent and automated analysis of urban soundscapes has a number of valuable applications. For example, it can be used to enhance context-aware computing, particularly for robotic navigation in changing urban environments including for autonomous vehicles (private, public transport, cargo), drones, robotic assistants, wheelchairs, or even tour guides [24, 25, 88, 94]. In these applications, sound analysis can be used to recognize and focus attention on sources outside the field of vision of autonomous devices, e.g., incoming traffic, emergency vehicles, someone calling; or to shape the system's response to contextual variables such as the terrain in which a robotic wheelchair is operating, or the soundscape level and composition to which an intelligent hearing aid needs to adjust.

These technologies can also contribute to content-based retrieval applications dealing with urban data, such as personal audio archiving [34], highlight extraction [93], video summarization [55], and searching through CCTV or mobile phone data [82]. In these scenarios sound analysis can help characterize patterns of similarity, novelty, anomaly, and recurrence in audio and multimedia content that can facilitate search and navigation.

However, there are two application domains in particular that are driving increased interest in automatic urban sound analysis: audio surveillance and noise pollution monitoring.

Audio Surveillance The need for automatic or semi-automatic surveillance in urban areas has experienced progressive and rapid growth, particularly in the past three decades. This is due to the increased threat posed by crime and terrorism. Surveillance systems were originally operated solely by humans, who had to constantly monitor video streams coming from the large number of cameras required to cover wide and complex areas of interest. In order to guarantee safety, however, full coverage of such areas would often require an unreasonably large number of operators. In addition, while it is difficult to outperform human monitoring with machines, this is only true when human attention is at its peak, which cannot be guaranteed over a lengthy period of time.

As a result, much effort has been devoted to the development of high-end technologies capable of alerting humans of potential hazards before they turn into a full-blown threat or calamity. Examples include the detection of fights/brawls [29, 53, 80] and intrusion [36, 97]. Technology improvements mean that infrared cameras for night-time operation have become affordable and less noisy; video

resolution can now guarantee an interocular distance of tens of pixels even from afar (for face recognition), and the dynamic range has grown to withstand the most adverse outdoor/indoor conditions. At the same time, signal processing for hazard detection has become more sophisticated, accommodating advanced illumination models; complex machine intelligence algorithms for video analytics; and advanced multimodal sensor fusion techniques, making fully automated surveillance systems effective and reliable enough to be fruitfully employed.

Many potentially dangerous events, however, can only be detected at an early stage through the analysis of an audio stream. Relevant examples range from the detection of specific sound sources such as gunshots, screams, and sirens; to actions like a car suddenly screeching to a stop; to scenes such as a brawl outside a night club, or a mugging. Audio surveillance is particularly beneficial in highly cluttered scenes, where visual events are likely to be occluded. Hence, the past decade has seen audio-based surveillance systems on the market, and new research focusing on the identification of dangerous events from the analysis of audio streams alone [27, 50, 69, 89] or from joint audio–video analysis [28, 96]. Crucially, sound event detection across dense sensor networks enables important surveillance capabilities such as localization and tracking of acoustic sources [9].

Noise Monitoring: Noise pollution is one of the topmost quality of life issues for urban residents worldwide [37]. In the United States alone, it has been estimated that over 70 million urban residents are exposed to harmful levels of noise [42, 62]. Such levels of exposure have proven effects on health such as sleep disruption, stress, hypertension, and hearing loss [8, 15, 43, 61, 90]. There is additional evidence of impact on learning and cognitive impairment in children [8, 14], productivity losses resulting from noise-related sleep disturbance [35, 92], and impact on real estate markets [63, 64].

Most major cities have ordinances that seek to regulate noise generation as a function of time of day/week and location. These codes define and measure noise in terms of overall sound pressure level (SPL) and its derivative metrics [87]. Such standards are in marked contrast with the emphasis on *sound sources* that is prevalent in noise surveys and complaints, as well as throughout the literature on the effect of noise pollution. The need for source-specific metrics is acknowledged by noise experts [87], especially in urban environments that are constantly reshaped by a large numbers of sources. As with audio surveillance, the benefits of applying sound classification technologies are evident and motivate recent efforts from the research community [68, 75–77].

The shortcomings of noise monitoring using SPL metrics are compounded by the difficulties of monitoring at scale. Site inspections by city officials are often few and far between and insufficient to capture the dynamics of noise across time and space. Alternatively, cities rely on civic complaint systems for noise monitoring such as New York City’s 311, effectively the largest noise reporting system anywhere in the world [66]. However, research shows that noise information collected by such systems can be biased by location, socio-economic status, and source type, failing to accurately characterize noise exposure in cities [65]. Therefore, recent years have

seen a proliferation of work on using dense networks of mobile or fixed acoustic sensors as an alternative and complementary solution to noise monitoring. In this context, sound analysis can contribute to the identification of specific sources of noise and their characteristics (e.g., level, duration, intermittence, bandwidth). This can in turn empower novel insights in the social sciences and public policy regarding the relationship of urban sound to citizen complaints, reported levels of annoyance, stress, activity, as well as health, economic and educational outcomes.

13.3 Acoustic Sensor Networks

13.3.1 Mobile Sound Sensing

In recent years consumer mobile devices, namely smart phones have seen rapid improvements in processing power, storage capacity, embedded sensors, and network data rates. These advances coupled with their global ubiquity have paved the way for a new paradigm in large-scale remote urban sensing: participatory sensing [18, 21]. The idea behind this approach is to utilize the sensing, processing, and communication capabilities of consumer smart phones to enable members of the public to collect and upload environmental data from their surroundings. This approach benefits from the use of existing infrastructure (sensing platform and cellular networks) meaning that deployment costs are effectively zero, provides unrivaled spatial coverage and also allows for the gathering of the subjective response to these environments, in situ. The drawbacks of this approach mainly lie in the low temporal resolution of its data resulting from the submission of short term measurements and the quality of the gathered data, as the model, physical, and handling conditions of the smart phones may not be consistent, resulting in aggregated environmental data of variable accuracy. A number of initiatives have sought to crowdsource sound and noise monitoring using mobile devices [31, 45, 56, 72, 73, 79, 81]. Their apps are typically limited to logging geo-located instantaneous SPL measurements. The EveryAware project [4, 10, 11] is an EU project intending to integrate environmental monitoring, awareness enhancement and behavioral change by creating a new technological platform combining sensing technologies, networking applications, and data-processing tools. One of its sub-projects is the WideNoise application, which allows for the compilation of noise pollution maps using participants' smart phones, including objective and subjective response data. In addition to this, they are examining the motivations for participation among their user base, as well as monitoring behavior change resulting from the access to personalized sound information. The OnoM@p project [41] follows some of the same goals and strategies of the above initiative. Notably, they attempt to address the issue of erroneous data through a cross-calibration technique between multiple device submissions, a welcome development for mobile noise sensing, with the caveat that it requires large-scale public adoption to be successful.

13.3.2 *Static Sound Sensing*

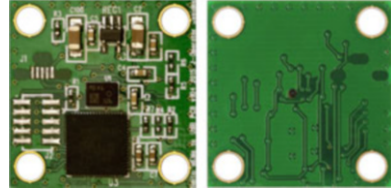
Static sound sensing solutions can take many forms with varying abilities and price points. Their main advantage over mobile sensing solutions is the ability to monitor continuously with increased levels of data quality. Highly accurate (± 0.7 dB), dedicated, commercially made networks such as the Bruel & Kjaer Noise Sentinel 3639-A/B/C [17] can produce legally enforceable acoustic data, but can cost upwards of \$15,000 USD per node. The high cost means that deployments are spatially sparse with durations usually in the order of a few months. Lower cost commercial solutions include the \$560 USD Libelium Waspote Plug & Sense Smart Cities device [52] which, amongst other things, measures decibel (dB) values with a ± 3.0 dB accuracy. The reduced cost per sensor node brings with it new possibilities for larger network deployments but a trade-off on data accuracy may mean its suitability is limited for large-scale urban deployments. Other examples [60] make use of hybrid deployments of low-cost, low-accuracy sensors with higher-cost, higher-accuracy sensors in an attempt to strike a balance between accuracy and scalability. Networks utilizing even lower cost sensors at the \$150 USD per sensor price point provide the potential for more network scalability, but make sacrifices in sensor capabilities. Examples of these [12, 46] make use of low-power computing cores that limit their ability to carry out any advanced in situ audio processing. With these acoustic sensor networks, it is desirable to have low-cost, powerful sensor nodes able to support the computational sound analysis techniques described in this book. The rest of this section will present the design and implementation of an acoustic sensor network capable of satisfying the cost, accuracy, and performance considerations described above.

13.3.3 *Designing a Low-Cost Acoustic Sensing Device*

In this section we describe the design of an acoustic sensing device developed in the context of the SONYC project,² a research initiative concerned with novel smart city solutions for urban noise monitoring, analysis, and mitigation. The device is based around the popular Raspberry Pi single-board computer (SBC) outfitted with a custom USB microelectromechanical systems (MEMS) microphone module where low-cost, acoustic accuracy, and high processing power are the primary considerations.

²<https://wp.nyu.edu/sonyc/>.

Fig. 13.1 Acoustic sensing module—back of board on *left* with MEMS microphone in center, front of board on *right* with microphone port in center



13.3.3.1 Microphone Module

In recent years, interest in microelectromechanical systems (MEMS) microphones has expanded due to their versatile design, greater immunity to radio frequency interference (RFI) and electromagnetic interference (EMI), low-cost and environmental resiliency [6, 7, 91]. Current MEMS models are generally 10× smaller than their more traditional electret counterparts. This miniaturization has allowed for additional circuitry to be included within the MEMS housing, such as a pre-amp stage and an ADC to output digitized audio in some models. The production process used to manufacture these devices also provides an extremely high level of part-to-part consistency, making them more amenable to multi-capsule and multi-sensor arrays. The sensing module shown in Fig. 13.1 uses an entirely digital design, utilizing a digital MEMS microphone (including a built-in ADC), and an onboard micro controller (MCU) enabling it to connect directly to the nodes computing device as a USB audio device. The digital MEMS microphone features a wide dynamic range of 32–120 dBA, ensuring all urban sound pressure levels can be effectively monitored. The use of an onboard MCU also allows for efficient, hardware level filtering of the incoming audio signal to compensate for the frequency response of the MEMS microphone before any further analysis is carried out. The standalone nature of this acoustic sensing module also means it is computing core agnostic, as it can be plugged into any computing device.

13.3.3.2 Form Factor, Cost, and Calibration

The sensor's prototype housing and form factor is shown in Fig. 13.2. The low-cost unfinished/unpainted aluminum housing was chosen to reduce radio frequency interference (RFI) from external sources, solar heat gain from direct sunlight and it also allows for ease of machining. All of the sensor's core components are housed within this rugged case except for the microphone and Wi-Fi antenna which is externalized for maximum signal gain.

In the prototype node shown in Fig. 13.2, the MEMS microphone is mounted externally via a repositionable metal goose-neck allowing the sensor node to be reconfigured for deployment in varying locations such as building sides, light poles, and building ledges. Figure 13.2 also shows the sensors bird spikes to ensure no damage is caused by perching birds. The total cost of the sensor excluding construction and deployment costs is \$83 USD, as of December 2016.

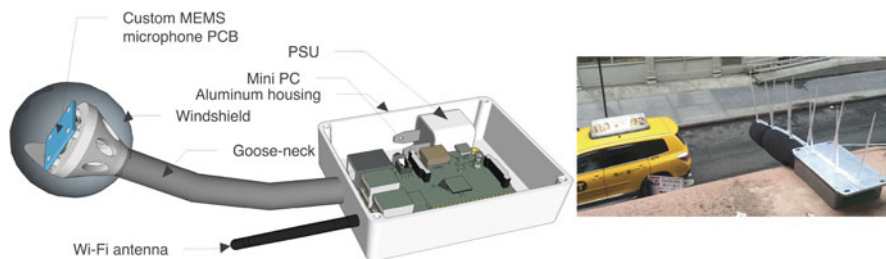


Fig. 13.2 Acoustic sensor node showing core components viewed from the underside (*left*) and a deployed node in NYC (*right*)

The sensing module was calibrated using a precision grade sound level meter as reference (Larson Davis 831 [49]), under low-noise, anechoic conditions. The sensor was then shown empirically to produce continuous decibel data at a level of accuracy used by the NYC city agencies tasked with enforcing the city’s noise code.

13.3.4 Network Design & Infrastructure

The prototype node relies on a continuous power supply and wireless network connectivity so its deployment locations are mainly determined by these prerequisites. Security and wider localized spatial acoustic coverage of sensors is maintained by mounting at a height of ~ 4 m above street level with a distance between sensors of around 2 city blocks or ~ 150 m. Ideally acoustic sensors would be mounted on poles, rather than on or close to building sides to reduce variations in SPL response due to wall proximity. Partnering with infrastructure owners/managers is crucial when selecting and deploying sensor nodes, and it is worth noting that the cost of deploying a sensor on urban locations such as light poles can spiral when lifting equipment and professional personnel are involved. Selection of sites with the likelihood of high variation in sound sources is also prioritized in order to facilitate the collection of a wide variation of ground-truth audio data as discussed in Sect. 13.4. In order to maintain public privacy, audio data is captured, losslessly FLAC³ compressed, and encrypted in 10 s snippets, interleaved with random durations of time. This data is transmitted from the sensor via Wi-Fi, directly to the project’s control server, which in turn transfers the data to the storage servers, ready for further analysis. Each sensor also transmits its current state every minute via a small “status ping”. This allows for near real-time remote telemetry display of all deployed sensors for fault diagnosis. Further in-depth control and maintenance of the deployed sensors is provided via a Virtual Private Network (VPN) that provides a method for remote Secure Shell (SSH) access to each node.

³<https://xiph.org/flac/>.

The VPN also enhances the wireless transmission security of the sensor as all data and control traffic is routed through this secure network. Future versions of the project's acoustic network will utilize multi-hop mesh networking approaches for sensor-server communications in order to increase the range of the network and reduce its power consumption to open up the possibility of battery powered, energy harvesting acoustic sensor nodes. Without the requirement of continuous power and pre-existing wireless network infrastructure, many more urban deployment possibilities become available.

13.4 Understanding Urban Soundscapes

Most prior work on understanding urban soundscapes has been focused on identifying acoustic scenes that are commonly found in urban environments such as parks, commercial streets, residential streets, construction sites, restaurants, or different modes of transportation (e.g., inside a taxi, train or bus). However, it is difficult to disambiguate work specific to urban environments from general acoustic scene classification (ASC) as described in Chap. 8. This is because the most widely used datasets for ASC research are largely or exclusively made from urban soundscapes. To make this clear, we provide a summary of those datasets in Table 13.1, where for each dataset we list the total number of audio recordings, the number of classes (acoustic scenes) and the number of these classes that can be considered urban sound scenes. As can be seen, all datasets contain a significant proportion of urban sound scenes.

While the focus of these datasets (and the approaches evaluated on them) is not necessarily urban sound scene analysis, they serve as a good proxy for it. Thus if we wish to understand the current state of the art in urban sound scene classification, we can refer to the DCASE 2016 acoustic scene classification challenge,⁴ which was based on the TUT Acoustic Scenes 2016 dataset [59] listed in Table 13.1. The challenge received close to 50 submissions spanning a variety of techniques,

Table 13.1 Some commonly used datasets for acoustic scene classification

Dataset	Recordings	Total scenes	Urban scenes
UAE noise DB series 1 [84]	10	10	9
UAE noise DB series 2 [84]	35	12	11
DCASE 2013 [38]	100	10	10
DARES G1 [40]	123	28	25
TUT acoustic scenes 2016 [59]	1170	15	13
LITIS rouen [71]	3026	19	19

As seen from the table, all datasets contain a significant proportion of urban scenes

⁴<http://www.cs.tut.fi/sgn/arg/dcase2016/>.

ranging from a baseline system which uses MFCC features with a GMM classifier, to deep learning architectures including fully connected and convolutional neural networks trained on a variety of input representations. Since the general problem of scene classification and the DCASE challenge are discussed in detail in previous chapters, here we will only limit ourselves to point out that the maximum reported classification accuracy was of 0.897, with incremental differences of 1% to the second and third best performing systems, and that the best performing method in the challenge was based on the late fusion of a deep and a shallow feature learner [33]. For a detailed comparison of algorithmic performance and further details about all participating methods the reader is referred to the challenge's results page.⁵

The challenge supports the notion that current strategies are already capable of providing robust solutions to urban ASC. This is not new, since high performance in this task has been reported for close to a decade at the time of writing [5]. At the same time, practically all datasets used for ASC evaluation to date are closed-set, meaning the data are divided into a fixed, known number of scenes. In a real-world scenario (for instance, a robot operating in a new environment) it is possible to encounter previously unheard acoustic scenes, which a model would have to identify as "unknown". Existing models are not trained to perform this task, which requires open-set data for training, and it is quite possible that model performance on this (more challenging) scenario would be lower.

Next we turn our attention to the more challenging task of sound source identification, which has received less attention and has ample room for improvement. As was the case before, this task is covered in detail elsewhere in this book, which is why for the rest of this section we will focus on research specifically targeting urban environments.

13.4.1 *Urban Sound Dataset*

In Chap. 6 a number of annotated datasets for environmental sound event detection and classification were discussed. While some of these contain sound events from urban soundscapes, up to 2013 there was no dataset focusing specifically on urban sounds. Previous work has focused on audio from carefully produced movies or television tracks [19], from specific environments such as elevators or office spaces [39, 70], and on commercial or proprietary datasets [23, 44]. The large effort involved in manually annotating real-world data means datasets based on field recordings tend to be relatively small (e.g., the event detection dataset of the IEEE AASP Challenge [39] consists of 24 recordings per each of 17 classes). A second challenge faced by the research community was the lack of a common vocabulary when working with urban sounds. This meant the classification of sounds into semantic groups varied from study to study, making it hard to compare results.

⁵<http://www.cs.tut.fi/sgn/arg/dcase2016/task-results-acoustic-scene-classification>.

Specific efforts to describe urban sounds have often been limited to subsets of broader taxonomies of acoustic environments (e.g., [16]), and thus only partially fulfill the needs of systematic urban sound analysis. To address this, Salamon et al. proposed an urban sound taxonomy [77] based on the subset of the taxonomy proposed by Brown et al. [16] dedicated to the urban acoustic environment. This taxonomy defines four top-level groups: human, nature, mechanical, and music, which are common in the literature [67], and specifies that its leaves should be sufficiently low-level to be unambiguous—e.g., car “brakes,” “engine,” or “horn,” instead of simply “car.” Furthermore, it is built around the most frequently complained about sound categories and sources—e.g., construction (e.g., jackhammer), traffic noise (car and truck horns, idling engines), loud music, air conditioners and dog barks—according to 370,000 noise complaints filed through New York City’s 311 service from 2010 to 2013.⁶

A subset of the resulting taxonomy, focused on mechanical sounds, is provided in Fig. 13.3. A scalable digital version of the complete taxonomy is available online.⁷ Rounded rectangles represent high-level semantic classes (e.g., human, nature, mechanical, music). The leaves of the taxonomy (rectangles with sharp edges) correspond to classes of concrete sound sources (e.g., siren, footsteps). For conciseness, leaves can be shared by several high-level classes (indicated by an earmark).

From this taxonomy, a dataset [77] was developed by focusing on ten low-level classes: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gunshot, jackhammer, siren, and street music. With the exception of “children playing” and “gun shot” which were added for variety, all other classes were selected due to the high frequency in which they appear in NYC urban noise complaints.

The audio data was collected from Freesound,⁸ an online sound repository containing over 160,000 user-uploaded recordings under a creative commons license. For each class, the authors downloaded all sounds returned by the Freesound search engine when using the class name as a query (e.g., “jackhammer”), manually inspected all recordings and kept only actual urban field recordings where the sound class of interest was present, and used Audacity⁹ to label the start and end times of every occurrence of the sound in each recording, with an additional *salience* description indicating whether the occurrence was subjectively perceived to be in the foreground or background of the recording. This resulted in a total of 3075 labeled occurrences amounting to 18.5 h of labeled audio. The distribution of total occurrence duration per class and per salience is provided in Fig. 13.4a.

The resulting dataset of 1302 full and variable length recordings with corresponding sound occurrence and salience annotations, *UrbanSound*, is freely available

⁶<https://nycopendata.socrata.com/data>.

⁷<http://serv.cusp.nyu.edu/projects/urbansounddataset/>.

⁸<http://www.freesound.org>.

⁹<http://audacity.sourceforge.net/>.

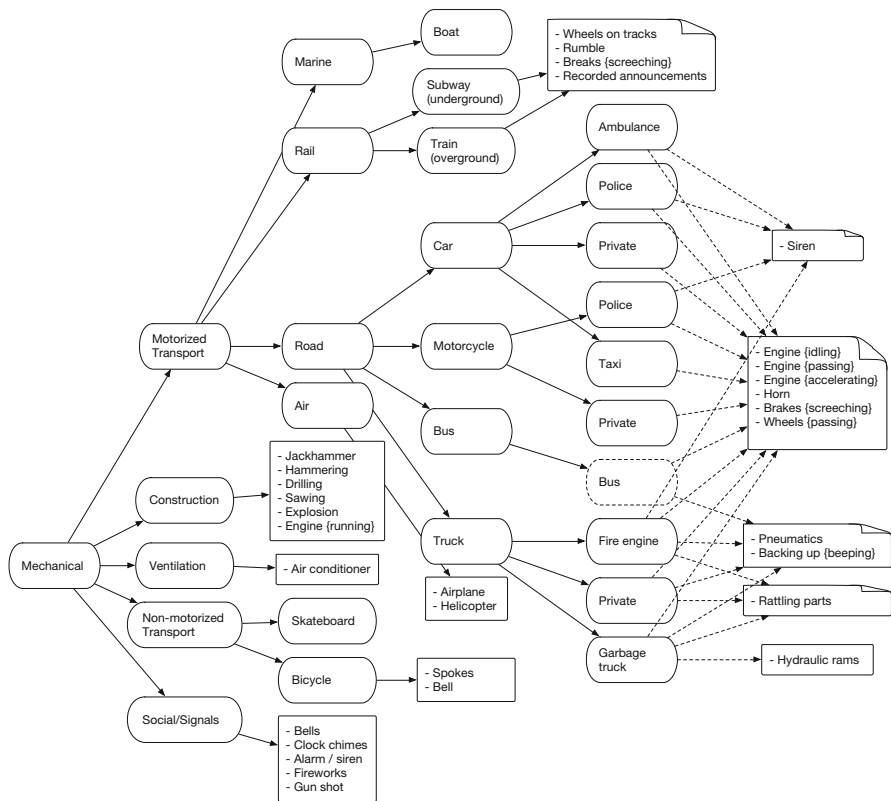


Fig. 13.3 Subset of the Urban Sound Taxonomy [77] focusing on mechanical sounds

online.¹⁰ Moreover, for research on sound source classification the authors curated a subset of short audio snippets, the *UrbanSound8K* dataset (also available online at the same url). Following the findings in [25], these snippets are limited to a maximum duration of 4 s. Longer clips are segmented into 4 s clips using a sliding window with a hop size of 2 s. To avoid large differences in the class distribution, there is a limit of 1000 clips per class, resulting in a total of 8732 labeled clips (8.75 h). The distribution of clips per class in *UrbanSound8K* with a breakdown into salience is provided in Fig. 13.4b.

A number of signal processing techniques and machine learning models have been proposed to date for urban sound classification and evaluated on the *UrbanSound8K* dataset [68, 74–77]. In the following sections we will review and contrast these approaches, comparing their performance in terms of classification accuracy. A summary of the key characteristics of each approach is provided in Table 13.2.

¹⁰<http://serv.cusp.nyu.edu/projects/urbansounddataset/>.

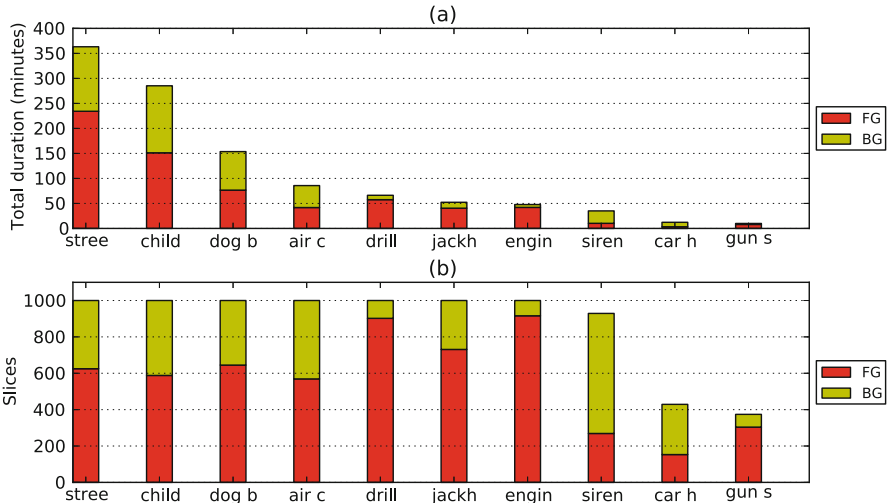


Fig. 13.4 (a) Total occurrence duration per class in UrbanSound. (b) Clips per class in UrbanSound8K. Breakdown by foreground (FG)/background (BG)

Table 13.2 Methods for urban sound classification

Method	Input features	Model
Baseline [77]	MFCC summary statistics	SVM
SKM-mel [75]	Dictionary encoded log-mel-spectrogram	Random forest
SKM-scattering [74]	Dictionary encoded deep scattering spectrum	SVM
Piczak-CNN [68]	Log-mel-spectrogram + delta	Deep CNN
SB-CNN [76]	Log-mel-spectrogram	Deep CNN + augmentation

13.4.2 Engineered vs Learned Features

The first step employed by all methods listed in Table 13.2 is feature extraction, i.e., transforming the raw audio signal into a feature space that is more amenable to machine learning. We can group audio feature spaces into two broad categories: designed (or engineered) features, and learned features. The former includes all features whose computation is independent of the input data, i.e., they are defined as the concatenation of operations whose goal is to capture a certain characteristic of the audio signal. The latter category includes features spaces that are learned directly from the data, including, for example, dictionary learning and deep learning methods.

Audio classification systems, including methods for environmental sound source classification, have traditionally relied on engineered features [19, 44, 70]. Thus the baseline system listed in Table 13.2 is a combination of a popular feature, the Mel-Frequency Cepstral Coefficients (MFCC), and a standard classification model

(Random Forest). However, most recent methods, including the remainder of the methods listed in the table, fall under the category of feature learning.

The first method listed following the baseline, SKM-mel [75], is based on unsupervised dictionary learning. The idea is to learn a dictionary of representative codewords directly from the audio signal in a data-driven fashion. The learned dictionary is then used to encode the samples in a dataset into feature vectors, which are then used to train/test a discriminative model of choice. The method employs the *spherical k-means* algorithm (SKM [26]) to learn the dictionary. Unlike the traditional k-means clustering algorithm [54], the codewords are constrained to have unit L2 norm (they must lie on the unit sphere, preventing them from becoming arbitrarily large or small), and represent the distribution of meaningful directions in the data. Compared to standard k-means, SKM is less susceptible to events carrying a significant amount of the total energy of the signal (e.g., background noise) dominating the dictionary. The algorithm is efficient and highly scalable, and it has been shown that the resulting set of vectors can be used as a dictionary for mapping new data into a feature space which reflects the discovered regularities [26, 30, 86]. The algorithm is competitive with more complex (and consequently slower) techniques such as sparse coding and has been used successfully to learn features from audio for music [32], and birdsongs [86]. After applying this clustering to the training data, the resulting cluster centroids can be used as the codewords of the learned dictionary. The number of codewords learned is typically much larger than the number of classes present in the data. It is also typically larger than the dimensionality of the input representation, i.e., the algorithm is used to learn an over-complete dictionary.

The clustering produces a dictionary matrix with k columns, where each column represents a codeword. Every sample in the dataset is encoded against the dictionary by taking the matrix product between each frame of its input representation, a mel-spectrogram, and the dictionary matrix. Every column i ($i = 1 \dots k$) in the resulting encoded matrix can be viewed as a time series whose values represent the match scores between the input representation and the i th codeword in the dictionary: when the input is similar to the codeword the value in the time series will be higher, and when it is dissimilar the value will be lower.

To ensure that all samples in the dataset are represented by a feature vector of the same dimensionality, the time series are summarized over the time axis by computing the mean and standard deviation of each time series and using these as features. The resulting feature vectors are thus all of size $2k$ and are standardized across samples before being passed on to the classifier for training and testing.

Note that for learning, one can choose to learn features from individual frames of the input representation, or alternatively group the frames into 2D patches and apply the learning algorithm to the patches. In [75] the authors show that the latter approach facilitates the learning of features that capture short-term temporal dynamics, which proves to be important for urban sound classification. The best result reported by the authors was obtained using patches with a time duration of roughly 370 ms (16 frames). For training, patches are extracted from the mel-spectrogram using a sliding window with a hop size of 1 frame. This results in

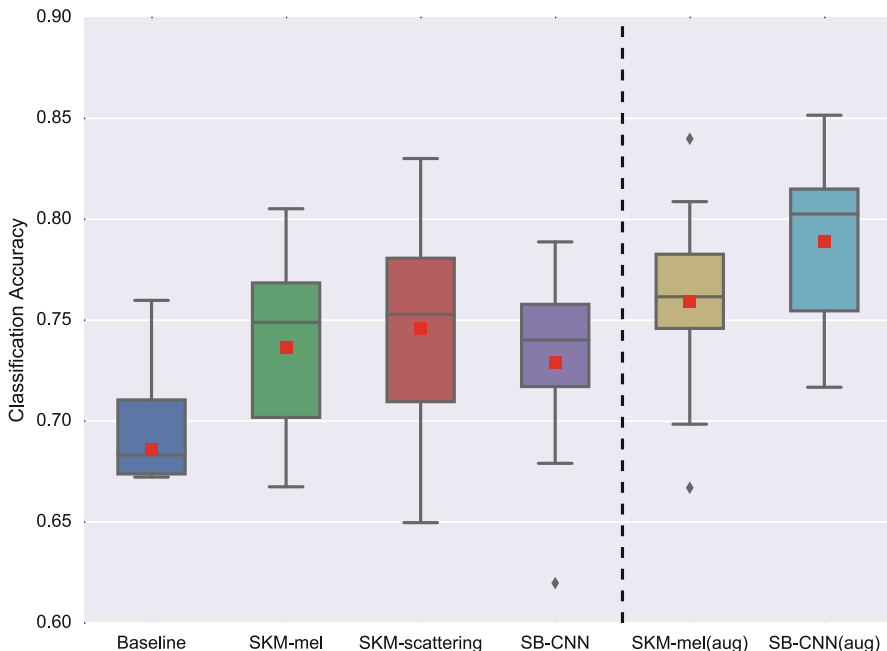


Fig. 13.5 Classification accuracy obtained on the UrbanSound8K dataset by different models: MFCC Baseline [77], spherical k-means dictionary learning from mel spectra [75] (SKM-mel), SKM learned from deep scattering spectra [74] (SKM-scattering) and the deep CNN proposed by Salamon and Bello [76] (SB-CNN). Models to the left of the dashed line were trained without data augmentation. To the right of the dashed line we present the results obtained by SKM-mel and SB-CNN when trained on an augmented training set: SKM-mel(aug) and SB-CNN(aug), respectively

significantly more training data for the unsupervised dictionary learning stage, and also ensures that the learned codewords account for different time-shifts of each sound source, hopefully increasing the robustness of the model to such shifts in the data.

While one could use the resulting patches directly as input for the feature learning, it has been shown that the learned features can be significantly improved by decorrelating the input dimensions using, e.g., Zero-phase Component Analysis (ZCA) whitening [47] or Principal Component Analysis (PCA) whitening [26].

Figure 13.5 presents classification accuracy results for UrbanSound8K in the form of a boxplot computed from the per-fold accuracies obtained by each model. Mean accuracies are indicated by the red squares. We will initially focus on the two left-most boxes and will discuss the remainder of the results in the following sections.

We clearly see that the SKM-mel model outperforms the MFCC baseline, with mean accuracies of 0.74 and 0.68, respectively. The difference is robust to the parameters of the mel-spectrogram, which are optimal for both reported results, but depends on the size of the dictionary for SKM, with best results for $k = 2000$ [75].

Such a significant improvement provides clear evidence of the advantage of feature learning compared to off-the-shelf engineered features, even when using a simple and shallow feature learning approach such as SKM.

13.4.3 Shift Invariance via Convolutions

The following method in Table 13.2, SKM-scattering [74], uses a different input representation altogether—the *scattering transform* [1–3]. This representation can be viewed as an extension of the mel-spectrogram that computes modulation spectrum coefficients of multiple orders through cascades of wavelet convolutions and modulus operators. Given a signal x , the first-order (or “layer”) scattering coefficients are computed by convolving x with a wavelet filterbank ψ_{λ_1} , taking the modulus, and averaging the result in time by convolving it with a low-pass filter $\phi(t)$ of size T :

$$S_1x(t, \lambda_1) = |x * \psi_{\lambda_1}| * \phi(t). \quad (13.1)$$

The wavelet filterbank ψ_{λ_1} has an octave frequency resolution Q_1 . By setting $Q_1 = 8$ the filterbank has the same frequency resolution as the mel filterbank, and this layer is approximately equivalent to the mel-spectrogram. The second-order coefficients capture the high-frequency amplitude modulations occurring at each frequency band of the first layer and are obtained by:

$$S_2x(t, \lambda_1, \lambda_2) = ||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi(t). \quad (13.2)$$

In [74] $Q_1 = 8$ and $Q_2 = 1$, the filterbank is constructed of 1D Morlet wavelets, and T is set to the same duration covered by the 2D mel-spectrogram patches used for dictionary learning in [75], i.e., 370 ms (for a sampling rate of 44,100 Hz this implies $T = 1024 \times 16$). Higher order coefficients can be obtained by iterating over this process, but it has been shown that for the chosen value of T , most of the signal energy is captured by the first- and second-order coefficients [3].

For each frame the first-order coefficients are concatenated with all of the second-order coefficients into a single feature vector. The second order coefficients are normalized using the previous order coefficients as described in [3]. From this point the process replicates the method described in the previous section: PCA whitening, dictionary learning using SKM, projection into the feature space, summarization and classification, in this case using a support vector machine (although the difference with a random forest is minimal). Therefore, the main difference between [75] and [74] is the addition of a *phase-invariant convolutional layer*, which is able to capture amplitude modulations in the input representation, in a time-shift invariant manner.

Figure 13.5 shows that learning a dictionary from scattering coefficients as opposed to the mel-spectrogram results in a relatively marginal improvement in classification accuracy (0.75 vs 0.74). Notably, the authors did observe a 5

percentage-point absolute improvement in the classification accuracy (i.e., +0.05) of masked sounds, i.e., sounds that were labeled by the annotators of the dataset as being in the background of the acoustic scene. This fits with findings in the sound perception and cognition literature showing that modulation plays an important role in sound segregation and the formation of auditory images [22, 57, 95], further motivating the exploration of deep convolutional representations, such as the scattering transform, for machine listening.

However, the most important finding is that the scattering transform’s inherent invariance to local time shifts allows for the comparable performance between SKM-scattering and SKM-mel, but using a dictionary that is an order of magnitude smaller ($k = 200$ versus $k = 2000$) while reducing the amount of 2D patches (samples) necessary for training by an order of magnitude too. In other words, shift invariance results in smaller machines trained with less data that are equally powerful, a finding that motivates further exploration using deep convolutional approaches.

13.4.4 *Deep Learning and Data Augmentation*

The last two methods in Table 13.2, Piczak-CNN [68] and SB-CNN [76], are based on deep (feature) learning [51]. This means that unlike the methods described above, here there are multiple feature learning layers including both fully-connected (like SKM) and convolutional (like scattering) layers, the feature learning is fully integrated with the classifier, and the machine is trained using supervised methods and a discriminative objective.

Since the two CNNs perform comparably when trained on the original Urban-Sound8K dataset (and only SB-CNN was evaluated both with and without data augmentation as discussed further below), for the remainder of the discussion we shall focus on SB-CNN as an instance of a deep learning model. SB-CNN takes log-scaled mel-spectrograms with 128 bands and a duration of 3 s as input to the network. Each 3 s spectrogram “patch” is Z-score normalized. The model is comprised of three convolutional layers interleaved with two pooling operations, followed by two fully connected (dense) layers. Notably, the convolutional layers of SB-CNN use a comparatively small receptive field of (5, 5) compared to the input dimensions of (128, 128). This is intended to allow the network to learn small, localized patterns, or cues, that can progressively build-up evidence for the presence/absence of specific sources even when there is spectro-temporal masking by interfering sources.

During training the model optimizes cross-entropy loss via mini-batch stochastic gradient descent [13]. Each batch consists of 100 patches randomly selected from the training data (without repetition). The model is trained using a constant learning rate of 0.01 and dropout [85] with probability 0.5 is applied to the input of the last two layers. L2-regularization is applied to the weights of the last two layers with a penalty factor of 0.001. The model is trained for 50 epochs with

a validation set used to identify the parameter setting (epoch) that achieves the highest classification accuracy. Prediction is performed by slicing the test sample into overlapping patches, making a prediction for each patch and finally choosing the sample-level prediction as the class with the highest mean output activation over all patches.

From Fig. 13.5 we see that SB-CNN, while outperforming the baseline, does not outperform its “shallow” SKM counterpart. This suggests that the UrbanSound8K dataset, despite being the largest dataset publicly available for urban sound classification, is not sufficiently large for the benefits of high-capacity, deep learning models to become apparent.

To address this limitation and increase the model’s robustness to intra-class variance, the authors also trained SB-CNN using *data augmentation*, that is, the application of one or more deformations to the training set which result in new, additional training data [48, 58, 83]. Assuming the deformations do not change the validity of the labels, augmentation aims to increase the model’s invariance to said transformations and thus generalize better to unseen data.

The authors applied four types of audio deformations: time stretching, pitch shifting, dynamic range compression, and the addition of background noise at different SNR, resulting in a training set an order of magnitude larger than the original UrbanSound8K. Augmentation was performed using the MUDA library [58]. After training SB-CNN with augmentation [Fig. 13.5: SB-CNN(aug)], the model significantly outperforms the SKM approach. Furthermore, we see that this improvement is not independent of the use of deep learning—training the SKM approach with augmentation [Fig. 13.5: SKM-mel(aug)] failed to improve as much. Increasing the capacity of the SKM model by increasing the dictionary size from $k = 2000$ to $k = 4000$ did not yield any further improvement either, even with the augmented training set. Instead, it is the combination of an augmented training set and the increased capacity and representational power of the deep learning model that results in this state-of-the-art performance.

13.5 Conclusion and Future Perspectives

In this chapter we have discussed intelligent acoustic sensing and analysis in the context of urban environments, particularly as one component of a larger trend towards smart city solutions. While we discuss a range of potential applications, we focus on two, audio surveillance and noise monitoring, that motivate new and exciting developments at the intersection of ubiquitous sensing and machine listening capabilities such as sound event detection, classification, localization, and tracking. These new technologies have the potential to improve the public safety and quality of life of urban residents.

In our discussion of acoustic sensor networks, we clearly favored the use of static over mobile sensing, and presented an example of a low-cost, high-quality solution intended for noise monitoring. However, the intended application greatly influences that choice: precise source localization and tracking is desirable but not necessary for noise monitoring, and the cyclic and seasonal nature of noise patterns means that off-network responses can be estimated by exploiting spatial correlations with other data types encoding information about, e.g., traffic, zoning, nightlife, construction, and tourist activity. On the other hand, audio surveillance requires relatively-dense arrays of sensors, something that is prohibitively expensive for static sensor networks, even for low-cost solutions such as the one presented in Sect. 13.3. One possibility is to deploy selectively and densely, as it is done for specific applications such as gunshot detection in neighborhoods with high gun crime incidences.¹¹ However, this is not applicable to surveillance scenarios (e.g., emergencies or terrorism) which are less predictable in space. Therefore, future developments will most likely require leveraging sensing from smart phones and other consumer-grade mobile devices, which in turn requires finding robust solutions to on-the-fly calibration, synchronization, and embedded computing that work well for acoustic data.

We devoted significant attention to the tasks of sound event detection and classification in cities. While the results are promising and much improvement has been accrued in a short period of time, there is still significant room for improvement and important challenges ahead. For example, one of the challenges of urban sound analysis is the heterogeneity of source types, a problem for which large-capacity models and ensemble methods might prove beneficial, as has been shown in acoustic scene classification [33] and bioacoustic classification [78]. However, current annotated datasets are small, include only a handful out of hundreds of possible sources, and are weakly labeled, meaning that comprehensive multi-source annotations are the exception rather than the norm. This hinders the ability to test such solutions.

Furthermore, real-world applications are intended to work on continuous audio streams, but many of the datasets discussed only contain snippets and thus fail to characterize the complex temporal dynamics of urban soundscapes. This scenario calls for the exploitation of longer temporal relationships, making the combination of convolutional and recurrent models an attractive direction for future research. These problems and solutions have been studied in the context of general environmental sound analysis (e.g., [20]), but remains to be explored for urban applications.

Finally, these sets only contain a small and arbitrary sample of the full range of acoustic conditions one might encounter in urban outdoor environments, and to which these systems are supposed to generalize. While data augmentation can help to a certain extent, future developments will be dependent on significant data collection from large-scale acoustic sensor networks, whether mobile or fixed. Encouraging developments include the recent launch of the YouTube-8M dataset

¹¹<http://www.shotspotter.com/>.

of tagged videos,¹² which contain a sizable and diverse sample of urban acoustic environments from mobile devices, and the ongoing deployment of audio sensor networks by various smart cities initiatives such as SONYC.¹³

References

1. Andén, J., Mallat, S.: Multiscale scattering for audio classification. In: 12th International Society for Music Information Retrieval Conference, Miami, pp. 657–662 (2011)
2. Andén, J., Mallat, S.: Scattering representation of modulated sounds. In: 15th DAFx, York (2012)
3. Andén, J., Mallat, S.: Deep scattering spectrum. *IEEE Trans. Signal Process.* **62**(16), 4114–4128 (2014)
4. Atzmueller, M., Becker, M., Doerfel, S., Hotho, A., Kibanov, M., Macek, B., Mitzlaff, F., Mueller, J., Scholz, C., Stumme, G.: Ubicon: observing physical and social activities. In: 2012 IEEE International Conference on Green Computing and Communications (GreenCom), pp. 317–324. IEEE, New York (2012)
5. Aucouturier, J., Defreville, B., Pachet, F.: The bag-of-frames approach to audio pattern recognition: a sufficient model for urban soundscapes but not for polyphonic music. *J. Acoust. Soc. Am.* **122**(2), 881–891 (2007)
6. Barham, R., Goldsmith, M., Chan, M., Simmons, D., Trowsdale, L., Bull, S.: Development and performance of a multi-point distributed environmental noise measurement system using mems microphones. In: Proceedings of the 8th European Conference on Noise Control (Euronoise 2009) (2009)
7. Barham, R., Chan, M., Cand, M.: Practical experience in noise mapping with a MEMS microphone based distributed noise measurement system. In: 39th International Congress and Exposition on Noise Control Engineering (Internoise 2010) (2010)
8. Basner, M., Babisch, W., Davis, A., Brink, M., Clark, C., Janssen, S., Stansfeld, S.: Auditory and non-auditory effects of noise on health. *The Lancet* **383**(9925), 1325–1332 (2014)
9. Baxter, K.C., Fisher, K.: Gunshot detection sensor with display. US Patent 7,266,045, 2007
10. Becker, M., Caminiti, S., Fiorella, D., Francis, L., Gravino, P., Haklay, M.M., Hotho, A., Loreto, V., Mueller, J., Ricchiuti, F., et al.: Awareness and learning in participatory noise sensing. *PLoS One* **8**(12), e81638 (2013)
11. Becker, M., Mueller, J., Hotho, A., Stumme, G.: A generic platform for ubiquitous and subjective data. In: Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, pp. 1175–1182. ACM, New York (2013)
12. Bell, M.C., Galatioto, F.: Novel wireless pervasive sensor network to improve the understanding of noise in street canyons. *Appl. Acoust.* **74**(1), 169–180 (2013)
13. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: 19th International Conference on Computational Statistics (COMPSTAT), Paris, pp. 177–186 (2010)
14. Bronzaft, A.L.: The effect of a noise abatement program on reading ability. *J. Environ. Psychol.* **1**(3), 215–222 (1981)
15. Bronzaft, A.: Neighborhood noise and its consequences. Survey Research Unit, School of Public Affairs, Baruch College, New York (2007)
16. Brown, A.L., Kang, J., Gjestland, T.: Towards standardization in soundscape preference assessment. *Appl. Acoust.* **72**(6), 387–392 (2011)

¹²<https://research.googleblog.com/2016/09/announcing-youtube-8m-large-and-diverse.html>.

¹³<https://wp.nyu.edu/sonyc/>.

17. Bruel & Kjaer Noise Monitoring Terminal Type 3639 (2015). <http://www.bksv.com/Products/EnvironmentManagementSolutions/UrbanEnvironmentManagement/NoiseInstrumentation/NoiseMonitoringTerminalFamily>
18. Burke, J.A., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S., Srivastava, M.B.: Participatory sensing. Center for Embedded Network Sensing (2006)
19. Cai, L.H., Lu, L., Hanjalic, A., Zhang, H.J., Cai, L.H.: A flexible framework for key audio effects detection and auditory context inference. *IEEE Trans. Audio Speech Lang. Process.* **14**(3), 1026–1039 (2006). doi:10.1109/TSA.2005.857575
20. Kahir, E., Heittola, T., Huttunen, H., Virtanen, T.: Polyphonic sound event detection using multi label deep neural networks. In: 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2015)
21. Campbell, A.T., Eisenman, S.B., Lane, N.D., Miluzzo, E., Peterson, R.A.: People-centric urban sensing. In: Proceedings of the 2nd Annual International Workshop on Wireless Internet, p. 18. ACM, New York (2006)
22. Carlyon, R.: How the brain separates sounds. *Trends Cogn. Sci.* **8**(10), 465–471 (2004)
23. Chaudhuri, S., Raj, B.: Unsupervised hierarchical structure induction for deeper semantic analysis of audio. In: IEEE ICASSP, pp. 833–837 (2013). doi:10.1109/ICASSP.2013.6637765
24. Chu, S., Narayanan, S., Kuo, C.C.J., Mataric, M.J.: Where am I? scene recognition for mobile robots using audio features. In: 2006 IEEE International Conference on Multimedia and Expo, pp. 885–888. IEEE, New York (2006)
25. Chu, S., Narayanan, S., Kuo, C.C.: Environmental sound recognition with time-frequency audio features. *IEEE Trans. Audio Speech Lang. Process.* **17**(6), 1142–1158 (2009). doi:10.1109/TASL.2009.2017438
26. Coates, A., Ng, A.Y.: Learning feature representations with K-means. In: *Neural Networks: Tricks of the Trade*, pp. 561–580. Springer, Berlin, Heidelberg (2012)
27. Cristani, M., Bicego, M., Murino, V.: On-line adaptive background modelling for audio surveillance. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004 (ICPR 2004), vol. 2, pp. 399–402. IEEE, New York (2004)
28. Cristani, M., Bicego, M., Murino, V.: Audio-visual event recognition in surveillance video sequences. *IEEE Trans. Multimedia* **9**(2), 257–267 (2007)
29. Cristani, M., Raghavendra, R., Bue, A.D., Murino, V.: Human behavior analysis in video surveillance: a social signal processing perspective. *Neurocomputing* **100**, 86–97 (2013)
30. Dhillon, I., Modha, D.: Concept decompositions for large sparse text data using clustering. *Mach. Learn.* **42**(1), 143–175 (2001)
31. D’Hondt, E., Stevens, M., Jacobs, A.: Participatory noise mapping works! an evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring. *Pervasive Mob. Comput.* **9**(5), 681–694 (2013)
32. Dieleman, S., Schrauwen, B.: Multiscale approaches to music audio feature learning. In: 14th ISMIR, Curitiba (2013)
33. Eghbal-Zadeh, H., Lehner, B., Dorfer, M., Widmer, G.: CP-JKU submissions for DCASE-2016: a hybrid approach using binaural i-vectors and deep convolutional neural networks. Technical report, DCASE2016 Challenge (2016)
34. Ellis, D.P.W., Lee, K.: Minimal-impact audio-based personal archives. In: 1st ACM workshop on Continuous Archival and Retrieval of Personal Experiences, New York, NY, pp. 39–47 (2004)
35. First report of the Interdepartmental Group on Costs and Benefits, Noise Subject Group: An economic valuation of noise pollution – developing a tool for policy appraisal. Department for Environment, Food and Rural Affairs (2008)
36. Foresti, G.: A real-time system for video surveillance of unattended outdoor environments. *IEEE Trans. Circuits Syst. Video Technol.* **8**(6), 697–704 (1998)
37. García, A.: *Environmental Urban Noise*. Wentworth Institute of Technology Press, Boston, MA (2001)

38. Giannoulis, D., Benetos, E., Stowell, D., Plumbley, M.D.: IEEE AASP challenge on detection and classification of acoustic scenes and events - public dataset for scene classification task. Technical report, Queen Mary University of London (2012)
39. Giannoulis, D., Stowell, D., Benetos, E., Rossignol, M., Lagrange, M., Plumbley, M.D.: A database and challenge for acoustic scene classification and event detection. In: 21st EUSIPCO (2013)
40. Grootel, M., Andringa, T., Krijnders, J.: DARES-G1: Database of annotated real-world everyday sounds. In: Proceedings of the NAG/DAGA Meeting 2009, Rotterdam (2009)
41. Guillaume, G., Can, A., Petit, G., Fortin, N., Palominos, S., Gauvreau, B., Bocher, E., Picaut, J.: Noise mapping based on participative measurements. *Noise Mapp.* **3**(1), 140–156 (2016)
42. Hammer, M.S., Swinburn, T.K., Neitzel, R.L.: Environmental noise pollution in the United States: developing an effective public health response. *Environ. Health Perspect.* **122**(2), 115–119 (2014)
43. Heinrich, U.R., Feltens, R.: Mechanisms underlying noise-induced hearing loss. *Drug Discov. Today Dis. Mech.* **3**(1), 131–135 (2006)
44. Heittola, T., Mesaros, A., Eronen, A., Virtanen, T.: Context-dependent sound event detection. *EURASIP J. Audio Speech Music Process.* **2013**, 1 (2013)
45. Kanjo, E.: Noiseply: a real-time mobile phone platform for urban noise monitoring and mapping. *Mob. Netw. Appl.* **15**(4), 562–574 (2010)
46. Kivelä, I., Gao, C., Luomala, J., Ihalainen, J., Hakala, I.: Design of networked low-cost wireless noise measurement sensors. *Sensors Transducers* **10**, 171 (2011)
47. Krizhevsky, A.: The ZCA whitening transformation. Appendix A of learning multiple layers of features from tiny images, Technical Report, University of Toronto (2009)
48. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS), pp. 1097–1105 (2012)
49. Larson Davis Model 831-NMS permanent noise monitoring system (2015). <http://www.larsondavis.com/Products/NoiseMonitoringSystems/PermanentNoiseMonitoringSystem>
50. Lecomte, S., Lengellé, R., C. Richard, C., Capman, F., Ravera, B.: Abnormal events detection using unsupervised one-class svm-application to audio surveillance and evaluation. In: 2011 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), pp. 124–129. IEEE, New York (2011)
51. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
52. Libelium smart cities board technical guide (2015). <http://www.libelium.com/development/waspmote/documentation/smart-cities-board-technical-guide/>
53. Lin, W., Sun, M., Poovendran, R., Zhang, Z.: Group event detection for video surveillance. In: 2009 IEEE International Symposium on Circuits and Systems, pp. 2830–2833. IEEE, New York (2009)
54. Lloyd, S.: Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **28**(2), 129–137 (1982)
55. Ma, Y.F., Lu, L., Zhang, H.J., Li, M.: A user attention model for video summarization. In: 10th ACM International Conference on Multimedia, pp. 533–542 (2002)
56. Maisonneuve, N., Stevens, M., Ochab, B.: Participatory noise pollution monitoring using mobile phones. *Inf. Polity* **15**(1), 51–71 (2010)
57. McAdams, S.: Spectral fusion, spectral parsing and the formation of auditory images. Ph.D. thesis, Stanford University, Stanford (1984)
58. McFee, B., Humphrey, E., Bello, J.: A software framework for musical data augmentation. In: 16th International Society for Music Information Retrieval Conference, pp. 248–254. Malaga, Spain (2015)
59. Mesaros, A., Heittola, T., Virtanen, T.: TUT database for acoustic scene classification and sound event detection. In: 24th European Signal Processing Conference (EUSIPCO), Budapest (2016)
60. Mietlicki, F., Mietlicki, C., Sineau, M.: An innovative approach for long-term environmental noise measurement: Rumeur network. In: 10th European Congress and Exposition on Noise Control Engineering (EuroNoise), Maastricht (2015)

61. Muzet, A., et al.: The need for a specific noise measurement for population exposed to aircraft noise during night-time. *Noise Health* **4**(15), 61 (2002)
62. Neitzel, R.L., Gershon, R.R., McAlexander, T.P., Magda, L.A., Pearson, J.M.: Exposures to transit and other sources of noise among New York City residents. *Environ. Sci. Technol.* **46**(1), 500–508 (2011)
63. Nelson, J.P.: Airports and property values: a survey of recent evidence. *J. Transp. Econ. Policy* **14**, 37–52 (1980)
64. Nelson, J.P.: Highway noise and property values: a survey of recent evidence. *J. Trans. Econ. Policy* **16**, 117–138 (1982)
65. New York City Department of Health and Mental Hygiene: Ambient Noise Disruption in New York City, Data brief 45. New York City Department of Health and Mental Hygiene, NY (2014)
66. NYC 311 Website. <http://www1.nyc.gov/311/>
67. Payne, S.R., Davies, W.J., Adams, M.D.: Research into the Practical and Policy Applications of Soundscape Concepts and Techniques in Urban Areas. DEFRA, HMSO, London (2009)
68. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: 25th International Workshop on Machine Learning for Signal Processing (MLSP), Boston, MA, pp. 1–6 (2015). doi:10.1109/MLSP.2015.7324337
69. Rabaoui, A., Davy, M., Rossignol, S., Ellouze, N.: Using one-class svms and wavelets for audio surveillance. *IEEE Trans. Inf. Forensics Secur.* **3**(4), 763–775 (2008)
70. Radhakrishnan, R., Divakaran, A., Smaragdis, P.: Audio analysis for surveillance applications. In: *IEEE WASPAA'05*, pp. 158–161 (2005). doi:10.1109/ASPAA.2005.1540194
71. Rakotomamonjy, A., Gasso, G.: Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **23**(1), 142–153 (2015). doi:10.1109/TASLP.2014.2375575
72. Rana, R.K., Chou, C.T., Kanhere, S.S., Bulusu, N., Hu, W.: Ear-phone: an end-to-end participatory urban noise mapping system. In: *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 105–116. ACM (2010)
73. Ruge, L., Altakrouri, B., Schrader, A.: Soundofthecity-continuous noise monitoring for a healthy city. In: *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pp. 670–675. IEEE, New York (2013)
74. Salamon, J., Bello, J.P.: Feature learning with deep scattering for urban sound analysis. In: *2015 European Signal Processing Conference, Nice* (2015)
75. Salamon, J., Bello, J.P.: Unsupervised feature learning for urban sound classification. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane* (2015)
76. Salamon, J., Bello, J.P.: Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process. Lett.* **24**(3), 279–283 (2017)
77. Salamon, J., Jacoby, C., Bello, J.P.: A dataset and taxonomy for urban sound research. In: *22nd ACM International Conference on Multimedia (ACM-MM'14), Orlando, FL*, pp. 1041–1044 (2014)
78. Salamon, J., Bello, J.P., Farnsworth, A., Kelling, S.: Fusing shallow and deep learning for bioacoustic bird species classification. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans*, pp. 141–145 (2017)
79. Santini, S., Ostermaier, B., Adelman, R.: On the use of sensor nodes and mobile phones for the assessment of noise pollution levels in urban environments. In: *2009 6th International Conference on Networked Sensing Systems (INSS)*, pp. 1–8. IEEE, New York (2009)
80. Saxena, S., Brémond, F., Thonnat, M., Ma, R.: Crowd behavior recognition for video surveillance. In: *International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 970–981. Springer, Berlin, Heidelberg (2008)
81. Schweitzer, I., Meurisch, C., Gedeon, J., Bärtl, R., Mühlhäuser, M.: Noisemap: multi-tier incentive mechanisms for participative urban sensing. In: *Proceedings of the 3rd International Workshop on Sensing Applications on Mobile Phones*, p. 9. ACM, New York (2012)

82. Serizel, R., Bisot, V., Essid, S., Richard, G.: Machine listening techniques as a complement to video image analysis in forensics. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 948–952. IEEE, New York (2016)
83. Simard, P.Y., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: International Conference on Document Analysis and Recognition, vol. 3, Edinburgh, Scotland, pp. 958–962 (2003)
84. Smith, D., Ma, L., Ryan, N.: Acoustic environment as an indicator of social and physical context. *Pers. Ubiquit. Comput.* **10**(4), 241–254 (2006). doi:10.1007/s00779-005-0045-4. <http://dx.doi.org/10.1007/s00779-005-0045-4>
85. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**(1), 1929–1958 (2014)
86. Stowell, D., Plumbley, M.D.: Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. *PeerJ* **2**, e488 (2014). doi:10.7717/peerj.488. <http://dx.doi.org/10.7717/peerj.488>
87. Taber, R.: Technology for a quieter america, national academy of engineering. Technical report, NAEPR-06-01-A (2007)
88. Thrun, S., Bennewitz, M., Burgard, W., Cremers, A., Dellaert, F., Fox, D., Haehnel, D., Rosenberg, C., Roy, N., Schulte, J., et al.: Minerva: a second generation mobile tour-guide robot. In: IEEE International Conference on Robotics and Automation, pp. 3136–3141 (1999)
89. Valenzise, G., Gerosa, L., Tagliasacchi, M., Antonacci, F., Sarti, A.: Scream and gunshot detection and localization for audio-surveillance systems. In: IEEE Conference on Advanced Video and Signal Based Surveillance, 2007 (AVSS 2007), pp. 21–26 (2007)
90. Van Kempen, E., Babisch, W.: The quantitative relationship between road traffic noise and hypertension: a meta-analysis. *J. Hypertens.* **30**(6), 1075–1086 (2012)
91. Van Renterghem, T., Thomas, P., Dominguez, F., Dauwe, S., Touhafi, A., Dhoedt, B., Botteldooren, D.: On the ability of consumer electronics microphones for environmental noise monitoring. *J. Environ. Monit.* **13**(3), 544–552 (2011)
92. Wicke, L.: Die ökologischen Milliarden: das kostet die zerstörte Umwelt-so können wir sie retten. Kösel, Munich (1986)
93. Xu, M., Xu, C., Duan, L., Jin, J.S., Luo, S.: Audio keywords generation for sports video analysis. *ACM Trans. Multimed. Comput. Commun. Appl.* **4**(2), 1–23 (2008)
94. Yanco, H.A.: Wheellesley: a robotic wheelchair systems: Indoor navigation and user interface. In: Assistive Technology and Artificial Intelligence, pp. 256–268. Springer, Berlin, Heidelberg (1998)
95. Yost, W.: Auditory image perception and analysis: the basis for hearing. *Hear. Res.* **56**(1), 8–18 (1991)
96. Zajdel, W., Krijnders, J., Andringa, T., Gavrilă, D.: Cassandra: audio-video sensor fusion for aggression detection. In: IEEE Conference on Advanced Video and Signal Based Surveillance, 2007. AVSS 2007, pp. 200–205. IEEE, New York (2007)
97. Ziliani, F., Cavallaro, A.: Image analysis for video surveillance based on spatial regularization of a statistical model-based change detection. In: Proceedings of IEEE International Conference on Image Analysis and Processing, pp. 1108–1111. IEEE, New York (1999)

Part V
Perspectives

Chapter 14

Future Perspective

Dan Ellis, Tuomas Virtanen, Mark D. Plumbley, and Bhiksha Raj

Abstract This book has covered the underlying principles and technologies of sound recognition, and described several current application areas. However, the field is still very young; this chapter briefly outlines several emerging areas, particularly relating to the provision of the very large training sets that can be exploited by deep learning approaches. We also forecast some of the technological and application advances we expect in the short-to-medium future.

Keywords Audio content analysis • Sound catalogues • Sound vocabularies • Audio database collection • Audio annotation • Active learning • Weak labels • Applications of sound analysis

14.1 Introduction

The foregoing chapters of this book have provided a comprehensive view of the state of the art in sound scene and event recognition, ranging from perceptual and computational foundations, through core techniques and evaluation, to a series of relatively advanced application areas. However, this is a young field which is

D. Ellis
Google Inc, 111 8th Ave, New York, NY 10027, USA
e-mail: dpwe@google.com

T. Virtanen (✉)
Laboratory of Signal Processing, Tampere University of Technology, Tampere, Finland
e-mail: tuomas.virtanen@tut.fi

M.D. Plumbley
Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford,
Surrey GU2 7XH, UK
e-mail: m.plumbley@surrey.ac.uk

B. Raj
Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: bhiksha@cs.cmu.edu

developing rapidly. This chapter provides brief descriptions of emerging trends that did not appear in earlier chapters, as well as some brief speculation about the technologies and applications likely to be important in coming years.

Given the dominance of large-scale machine learning in so many fields, in the next section we revisit the question of obtaining the data with which to train such classifiers, including the problem of defining a vocabulary of labels to use, and some approaches to working with imperfectly labeled data. We then go on to briefly outline some applications and approaches we see on the horizon for this field.

14.2 Obtaining Training Data

In many fields, deep learning approaches (as described in Chap. 5) have shown startling abilities to rival human abilities to recognize images, words, etc. In all cases, however, the best performance has relied on a combination of massive computational power with enormous training data. Because the sound scene and event recognition community has a relatively short history, there are no well-established sources for these kinds of “big data” training sets in our field. This deficit is stimulating a variety of current research.

14.2.1 Cataloguing Sounds

A first step to automated content analysis of sound recordings is to compose a catalogue of sound classes to detect in them. The catalogue has two components. The first is the set of *exemplars* or *models* for each of the sound classes. The second is the *labels* we assign to the classes, and by which we refer to them.

In principle the labels could be arbitrary, e.g., random character strings that uniquely identify the sound class. In practice, the output of audio content recognition is generally intended for downstream automated or manual analysis, and it becomes convenient, or even necessary, for the labels to be semantically meaningful words or phrases that a human analyst can interpret. In effect, we must “name” the sounds in linguistic terms. It follows, therefore, that an essential part of building sound catalogues is to “know” how to label sounds in linguistic terms.

Since the sound labels are intended to appeal to the human sense of semantics, the simplest solution is to involve humans directly in the collection and labeling of sounds, an approach we refer to as *manual* sound event vocabulary creation. Eventually, though, in order to obtain a more comprehensive and current catalogue of sound labels, we will require semi- or fully automated techniques that can mine the web and other knowledge sources to create a list of sound labels, a process we refer to as *automatic* vocabulary creation.

14.2.1.1 Manual Sound Event Vocabularies

The most natural way to obtain the labeled data needed for supervised training is to directly solicit labels for snippets of sound from human annotators. Supervised classification generally requires a predefined vocabulary of possible labels, but collecting unconstrained labels (such as the free-text tags on www.freesound.org) can lead to very large vocabularies with many synonyms (distinct terms referring to essentially the same sound event). Thus, it is preferable to establish a predefined, fixed vocabulary and to constrain annotators to only use terms from the vocabulary. This, however, raises several issues when compared to completely free annotation: How to define the vocabulary, and how to make sure the annotators are familiar with all the categories.

Both these problems become more serious as the size and scope of the vocabulary grows. For the Urban Sounds taxonomy, Salamon et al. [38] chose to work with a limited set of 10 sound event classes (from “car horn” to “children playing”) that were both common and typical in their urban field recordings. These classes were intended to be representative, so the exact choices were not important, and there was no effort to cover all events. They then went through their raw data marking each instance of each event; raters could be expected to quickly learn the full set of 10 events and identify them without further prompting.

For the TUT Sound Events 2016 database, Mesaros et al. [30] asked annotators to mark every sound event they perceived with free-text labels specifying a noun (object) and verb (action). These labels were subsequently manually merged, yielding 18 sound event classes with adequate representation (at least 20–30 examples) to be used in classification experiments. The data were limited to two acoustic contexts, “Residential area” (7 event categories) and “Home” (11 event categories), making it possible to obtain reasonable coverage with few classes.

The AudioSet Ontology [15] adopts the ambitious goal of defining a sound event vocabulary that covers all environmental sounds at a uniform level of detail, manually constructed from seeds including WordNet [31] and “Hearst patterns” [16] applied to web text. The resulting hierarchy of around 600 events arranged under 7 top-level categories raises problems of making annotators aware of all the categories. Annotation was primarily achieved via “verification” (presenting annotators with a single or small set of candidate labels which were then marked as present or absent, but without asking whether other sounds were also present); there was some experimentation with an interactive labeling tool that allowed annotators to search for events whose definitions included provided keywords.

14.2.1.2 Automatic Creation of Sound Event Vocabularies

A larger problem is to automatically generate a comprehensive vocabulary of the set of all sound classes that one may expect to encounter in an audio recording. By “listing” sound classes, we mean the listing of their *labels*—words or phrases that represent distinct sounds (regardless of whether we actually have exemplars of these

sounds or not). Generating such a list may be used as a precursor to identifying or collecting associated sound samples, e.g., for strongly or weakly supervised learning of classifiers, or more generally as an indicator of the awareness of the existence of these sounds. Such a list will, however, be much longer than would be feasible using the manual procedures mentioned in Sect. 14.2.1.1, and will require automated methods to produce.

The obvious way to compose the vocabulary is to list sounds by name—i.e., the noun or noun phrase canonically designated to represent the sound. Unfortunately, while some sounds do have names, e.g., onomatopoeic terms such as “squeal” and “chirp,” etymologically rooted terms such as “music,” or explicitly designated names, e.g., “Beethoven’s ninth,” many more, possibly the majority, are referred to by characteristics of their manner of production. In effect, the labels applied to the sounds are “descriptors” rather than “names.”

Consequently, once past the named sounds, sound vocabularies must be extended either by *composing* the sound-descriptor phrases that act as sound labels using appropriate rules or by *mining* web corpora for them, or a combination of the two.

There are, however, several confounding factors. Sound-descriptor terms can refer to the *objects* that produce the sound (e.g., “airplane” or “wind chime”), the environment in which the sound is produced (e.g., “playground”), the mechanism that produces the sound (e.g., “sawing wood”) or to more complex characterizations of the entire sound-producing phenomenon (e.g., “metal scraping on concrete” or “children in a playground”). The words composing the labels may themselves have no direct implication of sound, e.g., the term “car idling” evokes a type of sound, but “idling” by itself is not immediately associated with sound (consider “man idling”). Naive construction of sound label lists will thus result in many spurious entries.

Vocabulary-generation mechanisms should therefore take a two-step approach: first generate *candidate* sound labels, and subsequently *filter* them to eliminate spurious candidates. For instance, as mentioned above, the AudioSet Ontology [15] generated candidates by applying modified “Hearst patterns” [16] to identify hyponyms of the word “sound,” but this set needed subsequent manual filtering. Säger et al. [37] generated a much larger set of candidate sound terms based on the principle that sound events arise from an object (i.e., a noun) engaged in a particular action (i.e., a verb) or in some specific state (i.e., an adjective). They collected a set of sound-relevant words including 1200 nouns, 40 verbs, and 75 adjectives, and composed all possible adjective–noun pairs (ANPs) and verb–noun pairs (VNPs). This overcomplete set was then pruned by keeping only the pairs that occurred as tag combinations for sound files on www.freesound.org, then further pruning away implausible results by rejecting combinations that relied on a single uploader, or that only occurred in conjunction with the same other pairs. This resulted in over 1000 sound concepts; Table 14.1 shows some of their examples.

Kumar et al. [26] took the pattern-based approach used by CMU’s never-ending language learner, “NELL” [32], to generate sound labels from the ClueWeb corpus [11]. They noted that sound-descriptor phrases can often be disambiguated based on whether they can be prefixed by the words “*sound of*” without changing their meaning. Consequently, by matching the template “sound(s) of <Y>” where *Y* is any

Table 14.1 Examples of detected “sound” ANPs and VNPs

Howling dog	Splashing water
Heavy rain	Howling wolf
Crackling footsteps	Echoing phone
Heavy metal	Extreme noise
Gurgling water	Breaking snow

Table 14.2 Patterns for discovering sound concepts in text

Pattern		Example concept
P1	(DT) VBG NN(S)	Honking cars
P3	(DT) NN(S) VBG	Dogs barking
P5	(DT) NN NN(S)	String quartet
P6	(DT) JJ NN(S)	Classical music

VBG is the part of speech tag for verbs in the gerund form, *NN* for nouns, *DT* for determiners, and *JJ* for adjectives

phrase of up to four words to identify candidate phrases, followed by the application of a rule-based classifier to eliminate noisy candidates, they obtained a list of over 100,000 sound labels. Table 14.2 shows some of their grammatical patterns along with representative matches. Further, by applying a classifier to features extracted from a dependency path between a manually listed set of acoustic *scenes* and the discovered sound labels, they were also able to discover ontological relations, for instance, that *forests* may be associated with the sounds of “birds singing,” “breaking twigs,” “cooing,” and “falling water,” and that *churches* are associated with “children laughing,” “church bells,” “singing,” and “applause.”

The solutions described so far only consider *contiguous* word sequences as candidates for sound labels. More generally, sound-describing phrases may also be *extracted* from longer noun or verb phrases in which not all constituents relate to the sound. For instance, “a cat runs past a dog mewling” has the constituent “cat mewling.” In preliminary experiments Pillai and Qazi [35] found that candidates may be formed by parsing sentences into their components and evaluating combinations of various constituents of the sentence. Subsequent classification of formed candidates using a support-vector machine applied to vector-space representations of the phrases derived from a neural network returned lists of sound labels that were judged to be more than 80% accurate through manual inspection.

The lists obtained by these techniques can be further refined by considering frequency of occurrence across different webpages, their co-occurrence or affinity with one another, and the *contexts* they occur in. For instance, Pillai and Qazi [35] found that phrases derived from sound-related Wikipedia pages had a much higher likelihood of being valid candidates than those obtained from the larger web. Eventually, however, the true test is whether these phrases can indeed be associated with audio recordings. Thus a key feature that may be tested is to determine the frequency with which the phrases co-occur with sound files, particularly in contexts where phrases co-occurring with sounds may be expected (e.g., on sites such as soundforge.org or YouTube, or on Wikipedia pages about sounds).

14.2.2 *Opportunistic Data Collection*

When deep learning approaches were being developed for image recognition, the need for large numbers of labeled examples was addressed by mining the many billions of online digital images for examples whose captions suggested they contained the desired object. Even if captioned images make up only a tiny fraction of the billion-plus photos uploaded every day, there is still a very good chance of finding labeled examples for any common subject, and very often those examples will be clear, well-composed pictures.

Sound events are different. There is no widespread culture of uploading brief recordings of specific sounds; the closest is www.freesound.org, which has only a few hundred thousand sound files total. However, videos, while perhaps a thousand times less numerous than photos, are still available in ample volumes, including a proportion with associated text and other metadata. Users are perhaps more likely to describe the objects in their videos instead of the sounds, but those sounds are often associated with specific objects.

Thus, by a series of assumptions (some strong, some more reliable), we can take soundtrack snippets from videos whose metadata makes us believe they could contain the particular sound event for which we are collecting examples. Our assumptions might not work out, in which case we have a snippet labeled positive for containing sound X which in fact does not; we call this the “noisy labeling” problem. Even if the sound event does occur, there may be a lot of uncertainty about *when* it occurs; for instance, a 2 min video whose title is “AWESOME GLASS SMASH” may contain only a single glass breaking sound lasting under a second; this uncertainty around event timing we call the “weak labeling” problem.

Hershey et al. [17] use YouTube metadata to assign labels to videos; these labels are both weak (each label is assumed relevant to the entire video, whereas in fact it may only relate to specific time ranges within the soundtrack) and noisy (the label inference may have assigned a label that is not relevant at all). Their 3000 labels (“song,” “motorcycle,” etc.) are oriented towards YouTube searching behavior and thus may not necessarily relate to sound events, but their results—obtaining a mean Average Precision of up to 0.2, where random guessing would give something less than 0.01—show that, overall, the problems of weak and noisy labels are less devastating than might be expected.

14.2.3 *Active Learning*

Since human annotators are the ultimate authority for labeling sound examples, there is a strong incentive to maximize the value obtained for the expense of human labeling by ensuring they are shown the most important examples. This is the idea behind *active learning*, which includes a human annotator within a machine learning “loop,” so that each new label provides maximum value to the automatic

system [8]. Thus, instead of gathering labels for a large number of examples that simply confirm the confident predictions of a classifier trained on existing labels, only examples that the current system is most likely to misclassify (i.e., those on the boundaries between decision regions) are labeled. As the new labels improve the classifier, this boundary will shift, and the examples selected for labeling will change. Another aspect of this approach is to reduce annotator load by asking for less-precise judgments when the system can automatically refine them given some initial guidance; this philosophy has been effectively applied in image object segmentation, where annotators simply clicked a single point within an object, leaving the system to infer the most likely bounding box of the clicked object [34].

14.2.4 Using Unsupervised Data

Obtaining labeled sound event data is difficult or at least expensive, but unlabeled audio is plentiful, favoring any method able to exploit it. In [19], Jansen et al. process a million YouTube soundtracks (about 5 years of audio) using an online clustering system to produce millions of clusters. Although labels are not used in creating the clusters, they show that the resulting clusters are correlated with labels for labeled items, meaning that the unlabeled data can be used to help “regularize” a classifier trained on a smaller amount of labeled data.

Unlabeled data can also be used as a source of candidates for annotation [45]. Since many sound classes are rare, simply annotating randomly selected sound excerpts will have a very inefficient yield. Instead, given a few positive examples, excerpts with high acoustic similarity (by some measure) can be prioritized for annotation. An acoustic similarity measure that better approximates human similarity judgments (such as the embedding layer of a trained classifier) will give a correspondingly more useful prioritization.

14.2.4.1 Training with Weak Labels

While unlabeled data may be used to organize audio, eventually labeled data are needed to train classifiers for sound events. Ideally, these data would comprise isolated or cleanly segmented recordings of the target sound events. As mentioned earlier, such “strongly” labeled data are hard to come by, since the effort required to produce them is considerable.

It is much easier and cheaper to obtain weaker labels that merely indicate whether a particular sound event is *present* within a recording, without specifying additional details, such as the precise location of the event or even the number of times it occurs. Such labels may be obtained through manual annotation, e.g., the Google AudioSet corpus [15] which provides weak labels for 10-s snippets of audio. Alternatively, the labels may be inferred from the metadata or text attached to a recording, from analysis of any accompanying video, etc.

The focus now shifts to how best to train sound event classifiers with such weakly labeled data. How do we train classifiers to similarly tag (weakly label) other similarly sized snippets? At a finer level, can we use the weak labels to actually infer the *location* of the target sounds within the training data itself? Can we develop detectors to find and localize instances of the target events in novel *test* data?

These questions are analogous to a well-studied problem in the machine-learning literature: *multiple-instance learning* (MIL) [4]. Within the MIL paradigm, data instances are assumed to be grouped into *bags*. It is assumed that only bag-level labels are available, which indicate whether a bag contains representatives of a target class or not. The MIL tasks are now to (a) learn to best classify other *bags* (to determine what classes are present within them), and (b) to learn to classify *individual instances*, including those in the training data bags themselves. A number of algorithms have been proposed for MIL including methods based on boosting [2], random forests [27], support-vector machines [1], and neural networks [47]. MIL has been successfully applied to a variety of tasks including image recognition [29], text categorization [23], drug activity prediction [46], and bioinformatics [5].

In the sound-classification framework, the analogue of a bag of instances is a weakly labeled recording. The recording can in turn be split into short temporal or time-frequency segments, e.g., by uniformly segmenting it into fixed-length sections, for instance, half a second or one second long. These would comprise the individual *instances* in the bags. MIL techniques can now be directly applied. Classifying the individual instances (segments) as belonging or not belonging to the target sound event will naturally also localize the event to within the granularity of the segments.

The earliest reported application of MIL to audio analysis was by Mandel and Ellis [28], who applied it to music. Musical labels are generally applied to artists, albums, or individual tracks. However, a label may not apply to the entirety of an album or a track, e.g., a track tagged as “saxophone” may contain segments that have no sax in them at all. Mandel and Ellis attempted to apply MIL to obtain finer-grained tags from the high-level labels, i.e., to tag individual segments (at 10-s granularity) within the music, and reported being able to do so with reasonable accuracy.

Briggs et al. [6] applied a variant of MIL known as *multiple-instance multiple label* (MIML) learning to identify bird sounds in recordings. In a typical natural recording of bird sounds many different birds can be heard. The labels on training data generally only identify all the birds heard in them, but do not (and often cannot) isolate the individual birds. The authors demonstrated that the MIML solution provides better accuracies than other methods at identifying all birds in a test recording. Their solution was, however, restricted to performing bag-level classification; they did not attempt to isolate individual bird calls in either the test or the training data.

Kumar and Raj [24, 25] reported one of the earliest applications of MIL to the problem of generic sound event detection, and proposed a variety of solutions based on different classifier formalisms including support-vector machines and neural networks. The primary task addressed in their work was that of learning to *detect*

and localize sound events from weakly labeled training data. As a byproduct, their solutions also obtained temporal localization of the sound events within the training data itself. They were able to achieve both classification performance and temporal localization of some sound event classes comparable to that achieved with *strongly* labeled training data over a small vocabulary of sound events.

Xu et al. [44] and Kong et al. [22] proposed an alternative to the MIL approach, which treats the problem of learning to classify from weak labels as one of learning to pay *attention* to the right training instances in each bag. Their solution uses a combination of two neural networks, one which determines the importance of each instance in a bag, expressed as a weight assigned to the instance, and a second which attempts to classify it. The bag-level classifier output is a weighted combination obtained by summing the instance-wise multiplication of the outputs of both neural networks over all the instances in the bag. Both networks are jointly trained to minimize bag-level error. They showed that the resulting classifier is not only able to achieve highly accurate recording-level classification of audio, but also able to accurately localize events within both training and test recordings.

All of these proposed solutions have limited scope; their efficacy has only been demonstrated on small datasets and vocabularies. More recently, DCASE has issued a large-scale challenge on learning to classify sound events from weakly labeled data [13] that greatly increases the size, if not the vocabulary of the datasets. The challenge is expected to generate increased interest in the problem of learning from weakly labeled data.

Other outstanding problems include that the weak labels are often noisy. For instance, the AudioSet corpus reports that many of the weak labels in their dataset are inaccurate, with the accuracy of the labels falling below 50% for some categories, even with human annotators. Correia et al. [10] propose MIL solutions for cases where a confidence in the annotation may be established; more generally, however, the problem of training from noisy weak labels remains a challenge. A secondary, associated problem lies with the annotation of *negative* bags. Weak labels generally only indicate the *presence* of target sounds in a recording. The *absence* of sound events in a recording is rarely, if ever, annotated. Thus, the bags used as negative exemplars are only *assumed* to be negative, and are not guaranteed to be so. MIL solutions for noisy labels generally focus on noisy positive labels; the issue of noise in negative labels has been less considered. These are among the issues that must be resolved for truly scalable solutions for training with weakly labeled data.

14.2.4.2 Exploiting Visual Information

Thanks to the proliferation of smart phones, there are now billions of people carrying devices able to make recordings of their everyday experiences in both audio and video modalities; the hundreds of hours of video uploaded every minute to YouTube and similar services present both an important application domain and a rich source of training material for automatic sound scene and event recognition

systems. But the fact that these environmental audio recordings also include simultaneous visual information is an opportunity not to be ignored. In particular, given the difficulties in constructing accurate labels for audio recordings, can we glean useful labels from the video channel?

As mentioned above, image recognition systems are already very powerful, so a natural idea is to use existing image classifiers to provide the labels for training sound classifiers. One problem is that the labels provided by the image classifier—the objects visible in the scene, or some global label for the scene depicted—are at best only related to the sound events we would want to detect. At worst, they can be unrelated, either because the sound sources are not in the field of view or perhaps because the video has had an unrelated soundtrack dubbed on. However, the potential for enormous training sets can counterbalance these potential weaknesses in the labels.

This line of thinking was neatly developed by Aytar et al. [3]. They trained an audio classifier to predict the classification of the corresponding visual frame using existing pre-trained visual object and scene classifiers; the internal representation of this classifier was then used to train a simple SVM classifier for audio scene and event recognition tasks, substantially outperforming the best published results which did not have the benefit of the large audio-visual training set they were able to exploit.

14.2.5 Evaluation Tasks

The astonishing progress in image classification bears a substantial debt to the existence of the ImageNet [36] dataset and the associated evaluations. ImageNet provided at least 1000 positive example images for 1000 object categories, giving enough data to support the training of high-performance, deep network classifiers, and a broad enough range of object categories to give a passable attempt at general-purpose recognition.

ImageNet was the inspiration for AudioSet, a collection of manually labeled 10 s excerpts from the soundtracks of YouTube videos, providing at least 100 examples for over 500 sound event categories. While still much smaller than ImageNet, it at least attempts to provide comprehensive coverage of sounds rather than being limited to the small, specialized subsets of sound events that have been used in evaluations to date such as CLEAR [41] and DCASE [30, 40]. A standard evaluation based around AudioSet may similarly emerge as the common standard to push forward sound event detection.

14.3 Future Perspectives

14.3.1 Applications

Audio classification promises powerful applications in “embedded” intelligent devices that can benefit from adapting to unpredictable environments, from smart phones to self-driving cars. One significant recent development in this category is the smart home assistant, pioneered Amazon’s Echo [43]. This kind of hands-free smart assistant naturally relies on sound input for control; currently, this is exclusively via speech commands, but it is natural for it to use other information available in the acoustic channel, including the kind of home surveillance applications presented in Chap. 12, and raising all the privacy issues discussed there.

Another promising application area is personal hearing devices. Hearing aids that adapt automatically to their environments have been under development since at least 2005 [7], but the past few years have seen the emergence of intelligent “hearables,” presented as augmented earphones that promise features such as automatic removal of unwanted noise while passing through important or desired sounds [12]. Given their extreme constraints of size and power consumption, today’s devices merely suggest the kinds of functionality that will become possible as technology improves.

14.3.2 Approaches

The current generation of acoustic recognizers, as typified by the DCASE evaluations, focuses on an explicit set of output categories—either scenes or specific sound events. Despite recent efforts to develop a complete “ontology” of sound events [15], this approach seems doomed since there is an unlimited variety of sounds and subcategories within sounds that might be distinguished. One trend in fields including text and vision analysis is to work with an “embedding space,” a moderately sized feature space (e.g., 128 dimensions) where each object or event is mapped to a point such that semantically similar objects are close together [18]. Such a representation is intrinsically continuous, supporting arbitrarily fine distinctions between similar objects. Classification is not required, but if desired it can be accomplished by a simple quantization of the space. The embedding space is conveniently obtained as the activation of an intermediate layer in a neural net, trained by any method ranging from classical supervised training on explicitly labeled examples through to “triplet loss” approaches that require only same/different labels for pairs of examples, leaving implicit the underlying classes [39].

A common problem in trained classifiers is mismatch between training and test data: When tested on data that is systematically different from the examples used for training, performance may be arbitrarily degraded. This kind of mismatch

can include things that human listeners subconsciously ignore, for instance, the difference between the same sound source recorded in differently sized rooms (i.e., room acoustics), or mixtures with different background noises. To achieve the goal of human-level robustness at recognizing sound events, we will either need to collect training sets that span all relevant combinations of sources and environments (which becomes exponentially expensive) or devise alternative approaches to achieving this kind of generalization. Work in speech recognition has attempted to identify acoustic features that are relatively invariant to acoustic variations [20], although the alternative solution of collecting speech in very many acoustic conditions has ultimately proven more successful.

The idea of “transfer learning” [33] is aimed at situations where there is substantial out-of-domain training data that can nonetheless contribute to a task. Embedding space representations can be used for this kind of transfer: an embedding trained on one set of sound events—in which data straddles a wide range of recording conditions—can provide an embedding providing some invariance to recording conditions; if the embedding preserves enough source-relevant information to discriminate classes in a new task, then a classifier trained on the embedding representation of a small set of in-domain examples may result in a classifier that “inherits” the invariance from the larger dataset.

Current classifiers achieve robustness to background noise primarily through training on noisy examples, so essentially it is the combined properties of target event and interference that are being recognized. However, as discussed in Chap. 3, human perception appears to analyze complex scenes into distinct representations of individually perceived sources. Such a source separation or “Computational Auditory Scene Analysis” [9, 42] approach is conceptually appealing: an independent process able to divide a complex mixture into multiple, noise-free source sounds (along the lines of the matrix factorization techniques described in Sect. 8.3.3.2) would make the job of a subsequent event recognizer much easier. In practice, however, it is unlikely that such ideal source separation can be achieved without incorporating prior knowledge about source characteristics, so some kind of combined source separation and recognition process (reminiscent of the joint estimation of multiple sources in [14] and [21]) may turn out to be the most successful approach to source separation, and ultimately to robust sound event detection as well.

Although we have considered the classification of sound scenes and sound events as distinct tasks, they are of course related: a sound scene is essentially defined as a particular combination of sound events. Ideally, these two tasks can be unified, with scene classification emerging as a judgment over the set of detected events, although this approach is, for the moment, unlikely to rival global classification applied to the raw features from the scene.

14.4 Summary

We can safely expect high-accuracy automatic sound scene and event recognition in the near future, and it will lead to new and valuable applications in interactive systems and archive management. Sound provides critical information for us as inhabitants of the real world, and our automatic systems must and will take advantage of that information at our behest. The chapters in this book have provided detail on the current state of the art in the technology and applications of environmental sound recognition, and we look forward to the exciting developments that will unfold in the coming years.

References

1. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. In: *Advances in Neural Information Processing Systems*, pp. 577–584 (2003)
2. Auer, P., Ortner, R.: A boosting approach to multiple instance learning. In: *European Conference on Machine Learning*, pp. 63–74. Springer, Berlin (2004)
3. Aytar, Y., Vondrick, C., Torralba, A.: Soundnet: learning sound representations from unlabeled video. In: *Advances in Neural Information Processing Systems*, pp. 892–900 (2016)
4. Babenko, B.: Multiple instance learning: algorithms and applications. Technical Report, Department of Computer Science and Engineering, University of California, San Diego (2008)
5. Bandyopadhyay, S., Ghosh, D., Mitra, R., Zhao, Z.: MBSTAR: multiple instance learning for predicting specific functional binding sites in microRNA targets. *Sci. Rep.* **5**, 8004 (2015)
6. Briggs, F., Lakshminarayanan, B., Neal, L., Fern, X.Z., Raich, R., Hadley, S.J., Hadley, A.S., Betts, M.G.: Acoustic classification of multiple simultaneous bird species: a multi-instance multi-label approach. *J. Acoust. Soc. Am.* **131**(6), 4640–4650 (2012)
7. Büchler, M., Allegro, S., Launer, S., Dillier, N.: Sound classification in hearing aids inspired by auditory scene analysis. *EURASIP J. Adv. Signal Process.* **2005**(18), 387845 (2005)
8. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. *J. Artif. Intell. Res.* **4**(1), 129–145 (1996)
9. Cooke, M., Ellis, D.P.: The auditory organization of speech and other sources in listeners and computational models. *Speech Commun.* **35**(3), 141–177 (2001)
10. Correia, J., Trancoso, I., Raj, B.: Adaptation of SVM for MIL for inferring the polarity of movies and movie reviews. In: *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pp. 258–264. IEEE, New York (2016)
11. Dalvi, B., Callan, J., Cohen, W.W.: Entity list completion using set expansion techniques. In: *Proceedings of the Nineteenth Text REtrieval Conference (TREC 2010)*. NIST, Gaithersburg MD (2011)
12. Doppler Labs: HearOne wireless smart earbuds (2017). <http://hereplus.me>
13. Elizalde, B., Raj, B., Vincent, E.: Large-scale weakly supervised sound event detection for smart cars (2017). <http://www.cs.tut.fi/sgn/arg/dc2017/challenge/task-large-scale-sound-event-detection>
14. Frey, B.J., Deng, L., Acero, A., Kristjansson, T.T.: ALGONQUIN: iterating laplace’s method to remove multiple types of acoustic distortion for robust speech recognition. In: *INTER-SPEECH*, pp. 901–904 (2001)
15. Gemmeke, J.F., Ellis, D.P.W., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: an ontology and human-labeled dataset for audio events. In: *IEEE ICASSP 2017, New Orleans (2017)*. <https://research.google.com/pubs/pub45857.html>

16. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th Conference on Computational Linguistics, vol. 2, pp. 539–545. Association for Computational Linguistics, Stroudsburg, PA (1992)
17. Hershey, S., Chaudhury, S., Ellis, D.P.W., Gemmeke, J., Jansen, A., Moore, R.C., Plakal, M., Sauros, R.A., Seybold, B., Slaney, M., Weiss, R.: CNN architectures for large-scale audio classification. In: IEEE ICASSP 2017, New Orleans (2017). <https://research.google.com/pubs/pub45611.html>
18. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507 (2006)
19. Jansen, A., Gemmeke, J.F., Ellis, D.P.W., Liu, X., Lawrence, W., Freedman, D.: Large-scale audio event discovery in one million youtube videos. In: IEEE ICASSP 2017, New Orleans (2017)
20. Kingsbury, B.E., Morgan, N., Greenberg, S.: Robust speech recognition using the modulation spectrogram. *Speech Commun.* **25**(1), 117–132 (1998)
21. Klapuri, A.: Multiple fundamental frequency estimation by summing harmonic amplitudes. In: ISMIR, pp. 216–221 (2006)
22. Kong, Q., Xu, Y., Wang, W., Plumbley, M.D.: A joint detection-classification model for audio tagging of weakly labelled data. *CoRR abs/1610.01797* (2016). <http://arxiv.org/abs/1610.01797>
23. Kotzias, D., Denil, M., De Freitas, N., Smyth, P.: From group to individual labels using deep features. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 597–606. ACM, New York (2015)
24. Kumar, A., Raj, B.: Audio event detection using weakly labeled data. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 1038–1047. ACM, New York (2016)
25. Kumar, A., Raj, B.: Weakly supervised scalable audio content analysis. In: 2016 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE, New York (2016)
26. Kumar, A., Raj, B., Nakashole, N.: Discovering sound concepts and acoustic relations in text. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, New York (2017)
27. Leistner, C., Saffari, A., Bischof, H.: Miforests: multiple-instance learning with randomized trees. In: Computer Vision—ECCV 2010, pp. 29–42 (2010)
28. Mandel, M.I., Ellis, D.P.: Multiple-instance learning for music information retrieval. In: ISMIR, pp. 577–582 (2008)
29. Maron, O., Ratan, A.L.: Multiple-instance learning for natural scene classification. In: ICML, vol. 98, pp. 341–349 (1998)
30. Mesaros, A., Heittola, T., Virtanen, T.: Tut database for acoustic scene classification and sound event detection. In: Signal Processing Conference (EUSIPCO), 2016 24th European, pp. 1128–1132. IEEE, New York (2016). http://www.cs.tut.fi/~mesaros/pubs/mesaros_eusipco2016-dcase.pdf
31. Miller, G.A.: WordNet: a lexical database for English. *Commun. ACM* **38**(11), 39–41 (1995)
32. Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B., Krishnamurthy, J., Lao, N., Mazaitis, K., Mohamed, T., Nakashole, N., Platanios, E., Ritter, A., Samadi, M., Settles, B., Wang, R., Wijaya, D., Gupta, A., Chen, X., Saparov, A., Greaves, M., Welling, J.: Never-ending learning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15) (2015)
33. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
34. Papadopoulos, D.P., Uijlings, J.R., Keller, F., Ferrari, V.: Training object class detectors with click supervision. In: Proceedings of IEEE Computer Vision and Pattern Recognition (CVPR). Honolulu, Hawaii (2017). ArXiv preprint arXiv:1704.06189
35. Pillai, R., Qazi, U.W.: Acoustic analysis of text (aat): Extracting sound out of words. QSIURP Research Report, Carnegie Mellon University Qatar (2016)

36. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
37. Sager, S., Borth, D., Elizalde, B., Schulze, C., Raj, B., Lane, I., Dengel, A.: AudioSentiBank: large-scale semantic ontology of acoustic concepts for audio content analysis. arXiv preprint (arXiv:1607.03766) (2016)
38. Salamon, J., Jacoby, C., Bello, J.P.: A dataset and taxonomy for urban sound research. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 1041–1044. ACM, New York (2014). https://serv.cusp.nyu.edu/projects/urbansounddataset/salamon_urbansound_acmmm14.pdf
39. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)
40. Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.D.: Detection and classification of acoustic scenes and events. *IEEE Trans. Multimedia* **17**(10), 1733–1746 (2015)
41. Temko, A., Malkin, R., Zieger, C., Macho, D., Nadeu, C., Omologo, M.: Clear evaluation of acoustic event detection and classification systems. In: International Evaluation Workshop on Classification of Events, Activities and Relationships, pp. 311–322. Springer, New York (2006)
42. Wang, D., Brown, G.J.: Computational Auditory Scene Analysis: Principles, Algorithms, and Applications. Wiley/IEEE Press, New York (2006)
43. Wikipedia: Amazon Echo (2017). https://en.wikipedia.org/wiki/Amazon_Echo
44. Xu, Y., Kong, Q., Huang, Q., Wang, W., Plumbley, M.D.: Attention and localization based on a deep convolutional recurrent model for weakly supervised audio tagging. CoRR abs/1703.06052 (2017). <http://arxiv.org/abs/1703.06052>
45. Zhao, S., Heittola, T., Virtanen, T.: Active learning for sound event classification by clustering unlabeled data. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2017)
46. Zhao, Z., Fu, G., Liu, S., Elokely, K.M., Doerksen, R.J., Chen, Y., Wilkins, D.E.: Drug activity prediction using multiple-instance learning via joint instance and feature selection. *BMC Bioinf.* **14**(14), S16 (2013)
47. Zhou, Z.H., Zhang, M.L.: Neural networks for multi-instance learning. In: Proceedings of the International Conference on Intelligent Information Technology, Beijing, pp. 455–459 (2002)

Index

A

accuracy, 170, 379
 as a function of computational cost, 348
acoustic and auditory features, 41, 43, 46, 50, 51, 54–57, 61, 72–96
acoustic correlates, 41–43, 49–52, 54, 56–58, 61
acoustic language model, 233
acoustic model, 15
acoustic parameters, 41–43, 49–52, 54, 56–58, 61
acoustic scene classification, 216, 382
acoustic sensor network, 379
acoustic topic models, 236
acoustic variability, 342
acoustics, 41–43, 49–52, 54, 56–58, 61
action perception, 56
activation function, 125
active learning, 406
analysis frame, 20
annotation, 18
anonymization, 365
APE curves, 356, 358
application-independent methodology, 358
applications
 examples of, 338
 in the smart home, 336–339
applied probability of error curves, *see* APE curves
area under curve, 172
attentional processes, 41, 42, 51, 53, 61
audio capture quality, 345
audio context recognition, 216
audio event recognition (AER), 336
audio events, 341

audio features, 285
audio file format, 283
audio fingerprint, 293
audio surveillance, 376
audio tagging, 15, 30
audio-based sound retrieval, 290
audiovisual object, 256
auditory filter, 47, 48, 50, 57, 61
auditory object, 41, 42, 50–52
auditory scene, 41–43
auditory scene analysis, 41, 42, 46, 51, 53
auditory sketch, 59, 60
automatic music transcription, 231
average precision, 171

B

back-propagation through time, 134
bag of frames, 217, 341
beamforming, 260, 306, 313
big O notation, 322
bioacoustics, 303
biodiversity, 304, 318
block mixing, 164
bottleneck DNN, 89
brightness, 49

C

calibration, 355, 358, 381
 versus discrimination, 355
canonical correlation analysis, 248, 256
categorization, 183
citizen science, 309
city agencies, 381

classification, 26
 classifier, 104
 closed set, 219
 cloud computing, 347

- privacy concerns around, 365

 co-factorization, 250
 coherence analysis, 41, 51–53, 60
 comparative evaluation, 355
 complexity, 322
 computational cost, 347

- comparison across algorithms, 348
- evaluation of, 347
- versus classification accuracy, 348

 computational power, 345

- limitation of, 346

 consent to data usage, 363
 content-based retrieval, 376
 context-aware computing, 376
 context-dependent sound event detection, 232
 continuous power, 381
 continuous sound recognition, 342, 343, 351
 convolution, 389
 convolutional neural networks, 390
 convolutive NMF, *see also* non-negative matrix factorization (NMF), 230
 copyright, 283
 covariance

- diagonal, 116
- isotropic, 116
- spherical, 116

 creating datasets, 159
 creative commons, 283
 cross-classification, 190, 206
 cross-correlation, 310, 312, 314, 323
 cross-entropy, 126
 cross-validation, 165
 crowdsourcing, 309, 378

D

data

- ownership of, 363
- personal, sensitive and confidential, 362
- security, 365

 data annotation, 154, 155
 data augmentation, 139, 164, 390, 391
 data collection, 34
 data controller, 362
 data mining, 308
 data processor, 362
 data protection, 359–365

- law, 360, 361

 data quality, 379

datasets, 382
 DCASE, 217, 224
 DCF, 356, 357
 decibel, 379
 decision trees, 112
 deep learning, 123, 227, 324, 390

- convolutional network, 129
- convolutional-dense, 137
- convolutional-recurrent, 138
- initialization, 126
- multi-layer perceptron, 124
- recurrent network, 133
 - bi-directional, 137
 - gated recurrent unit, 134
 - long short-term memory, 135

 deep neural networks, 227

- open set DNNs, 344

 delta features, 22
 deploying, 381
 deployment, 380
 DET curve, 345, 356, 358
 detection, 15, 26, 309
 detection error trade off curve, *see* DET curve
 dictionary learning, 386, 387
 Dirichlet process mixture, 119
 discrete cost function (DCF), 356, 357
 discrete Fourier transform (DFT), 20
 discrimination versus calibration, 355
 discriminative model, 107
 distance estimation, 306
 distance sampling, 306
 domain adaptation, 139
 domain-specific knowledge, 294
 duration

- densities, 342
- modeling, 342

 dynamic range, 380
 dynamic time warping (DTW), 315, 323

E

eavesdropping, 365
 ecoacoustics, 318, 327
 edge, processing on the, 365
 EER, 353, 356
 embedded, 90
 embedded processing, 346, 365
 embedding space, 411
 empirical risk minimization, 105
 enforceable acoustic data, 379
 ensemble methods, 140
 environmental audio recordings

- privacy of, 362

equal error rate (EER), 353, 356
 error rate, 173
 error rates
 quantitative, 353
 errors
 qualitative assessment of, 353
 types of, 352
 ethical issues, 359–365
 evaluation
 comparative, 343, 355
 in an open set framework, 343
 metrics, 342, 355
 of computational cost, 347
 subjective, 352, 354
 evaluation protocols, 175
 evaluation setup, 165
 exemplar theory, 187
 existing datasets, 162
 explicit-duration HMM (EDHMM), *see*
 also hidden Markov model, *see*
 also hidden semi-Markov model
 (HSMM), 225, 226
 explicit-duration Markov model, 317

F

F-score, 170, 342, 344, 353, 355, 358
 faceted search, 289
 factorial HMM, *see also* hidden Markov model
 (HMM), 228
 false negative, 167
 false positive, 167
 feature extraction, 20
 feature fusion, 246
 feature learning, 22, 217, 218, 325, 386
 feature map, 130
 feature selection, 89
 feature stacking, 22, 227
 features, 325
 features, acoustic, 217
 filter, 90
 fluctuation strength, 50
 frame blocking, 20

G

gating, 58
 Gaussian mixture model, 115
 priors, 118
 generalisation, 313
 generative models, 112
 Gestalt, 50, 52, 185
 global coherence field, 263

gradient descent, 127
 grammar, 233
 grouping, 42, 50, 52, 53

H

hardware, 346
 Hidden Markov model, 119
 discriminative, 121
 priors, 123
 hidden Markov model, 224, 228, 315
 hidden semi-Markov model, 225, 342
 hierarchical HMM, *see also* hidden Markov
 model (HMM), 233
 histogram-of-events, 235
 holistic perception, 185
 home automation, 336–337
 horse, 353

I

ImageNet, 7, 410
 in-situ audio processing, 379
 inference
 Bayesian, 114
 maximum a posteriori, 113
 maximum likelihood, 113
 inference of sensitive data, 363
 infrastructure, 381
 instance-based learning, 323
 interaural intensity difference, 259
 interaural phase difference, 259
 interaural time difference, 258
 intermediate statistics, 167
 Internet of things, 336
 interrupted sequences, 341

K

Kalman filter, 266
 kernel methods, 111
 knowledge transfer, 340

L

language model, 233
 late fusion, 252
 latency, 347
 LDA, 89
 learning process, 23
 licensing, 283
 limitation
 of audio capture quality, 345
 of computational power, 346

linear model, 108
 linguistics, 197
 localisation, 306
 locality-sensitive hashing (LSH), 323
 log-likelihood ratio cost function, 356, 357
 logistic function, 109
 logistic regression, 109
 loss function, 104
 loudness, 46–49, 52, 56, 57, 61
 low cost audio subsystems, 345

M

machine learning, 290
 macro averaging, 168
 marketing value, 337
 Markov model, 316
 Markov renewal process, *see also* point process, 234, 318
 material perception, 41, 54–56, 61
 mel-band energy, 21
 mel-frequency cepstral coefficients, 21, 217, 291
 MEMS microphone, 379
 metadata, 284
 metadata-based sound retrieval, 286
 metrics

- objective, 355
- subjective, 354

 micro averaging, 168
 micro controller, 380
 microphone

- array, 345
- quality, 345

 microphone array, 257, 306
 minimal features, 54, 57
 mitigation, 379
 mobile devices, 378
 mobile sound sensing, 378
 morphological filtering, 310
 multiple-instance learning, 408
 multi-class models, 110
 multi-condition training, 17
 multi-label classification, 30
 multi-source, 227
 multi-space densities, 342
 multidimensional representation, 41, 49, 51, 52
 multidimensional scaling, MDS, 49
 multigrams, 343
 multilabel, 227
 multimedia sharing, 279
 multimodal, 244
 multiview data, 244

N

nearest neighbors, 112, 293
 neural transduction, 43, 47, 48
 noise code, 381
 noise pollution, 377
 noise pollution maps, 378
 non-linearity, 125
 non-negative matrix factorisation (NMF), 323
 non-negative matrix factorization (NMF), 229
 normalized decision cost (NDC), 174

O

onset, 50
 onset detection, 221, 222
 ontology, 294
 open set, 219, 313

- optimization, 358
- sound recognition, 343, 358

 open-set classification, 307
 operation point, 355
 opinion scores, 354
 overfitting, 25

P

parallel factor analysis, 249
 participatory sensing, 378
 particle filter, 266
 patterns, long term temporal, 342
 PCA, 89
 perception, 41, 42, 48, 50, 52, 54, 56, 58–61
 periodic and non-periodic signals, 44
 peripheral auditory system, 42, 43, 46, 47, 49, 61
 personal data, 362
 pitch, 43, 45, 46, 48, 50, 53, 54, 57, 58, 61
 pitch tracking, 310, 325
 pitfalls in data collection, 161
 point process, 234
 polyphonic sound event detection, 227
 polyphony, 215, 230
 pooling, 130
 pre-processing, 19
 precision, 170, 342, 344, 355
 privacy, 359–365, 381

- of audio data, 361
- right to, 360

 privacy preserving algorithms, 365
 processing on the edge, 365
 profiling, 362
 propagation effects, 306
 prototype theory, 186

psychoacoustics, 41, 43, 49
 psychomechanics, 54, 57

Q

qualitative assessment, 353
 quality
 of audio capture channels, 345
 quantitative error rates, 353
 query by example, 293
 query expansion, 288

R

random forest, 227
 random projection, 325
 recall, 170, 342, 344, 355
 receiver operating characteristic curve, *see*
 ROC curve
 recognition, 25, 42
 recommendation system, 295
 rectified linear unit, 125
 regularization, 106
 relevance score, 288
 remote alerting, 351
 right to privacy, 360
 robustness, 327
 ROC curve, 172, 356
 Rosch, 186
 roughness, 50

S

scalability, 379
 scattering, 327
 scattering transform, 326, 389
 search, 90
 search engine, 285
 security, 382
 security of data, 365
 segmentation, 309
 segregation, 41, 42, 50, 52, 53, 60, 61
 self-organizing map, 295
 semi-Markov model, 317
 sensitive data, inference of, 363
 sensor networks, 379
 sensor telemetry, 381
 sharpness, 49
 short-time Fourier transform, 291
 short-time processing, 20
 similarity, 191, 198, 314
 single board computer, 379

single label classification, 30
 sinusoidal modeling, 310, 325
 size perception, 41, 54, 56, 61
 smart cities, 374, 391
 smart home
 applications, 336
 definition, 336
 smart-phones, 378
 soft-max, 126
 sorting tasks, 191
 sound classification, 15, 30
 sound event, 41, 42, 51, 53, 54, 61
 sound event detection, 31, 222
 sound event labels, 153
 sound identification, 41–43, 54, 56, 57, 60, 61
 sound pressure level, 377
 sound production, 41, 43, 45, 54, 56
 sound propagation, 41–43, 45, 61
 sound recognition, 41–43, 54, 56, 57, 60, 61
 sound scene classification, 216
 sound scene labels, 152
 sound scene recognition, 216
 sound sharing, 280
 sound source variation, 381
 source localization, 262
 source separation, 228, 261, 313, 412
 sparse features, 57, 59–61
 sparse representations, 57, 59–61
 spatial covariance matrix, 260
 spatially explicit capture–recapture (SECR),
 306
 speaker diarization, 232
 species classification, 310
 spectral centroid, 49
 spherical k-means, 387
 SSH, 381
 static sound sensing, 379
 statistical ecology, 306
 status ping, 381
 steered response power, 263
 stratification, 166
 subjective evaluation, 352
 by opinion scoring, 354
 subjective metrics, 354
 supervised learning, 22, 105
 support vector machine, 108
 one-class, 344
 surveillance, 254
 switching state space model (SSSM), 233
 system calibration, 355, 358
 system design, 34
 system evaluation, 34

T

tag cloud, 289
tag recommendation, 287
tagging system, 287
taxonomy, 202, 383, 402
technological research, 33
template matching, 222, 312
temporal modeling, 224, 342
tensor, 249
term frequency–inverse document frequency, 288
test set, 344
timbre, 46, 49, 50, 52, 58, 60, 61
time difference of arrival, 259
tones and noises, 44
transfer function, 125
transfer learning, 412
triplet loss, 411
true negative, 167
true positive, 167

U

underfitting, 25
universal background model (UBM), 313
unsupervised learning, 85, 295, 407
urban sound classification, 383, 385
urban sound dataset, 383, 384
urban soundscape, 374
use cases, 336–340
 in the smart home, 336–339

user experience, 350–359
user interface, 350
user opinion, 352

V

velocity perception, 41, 54, 56, 57, 61
video indexing, 253
visual features, 245
visual information, 409
visualisation, 321
Viterbi decoding, 231
vocabulary, 402
vocal learning, 308
vocal tract, 309
voice activity detection, 222
VPN, 381

W

wavelet filterbank, 389
weak labels, 236, 407
Wi-Fi, 380, 381
Wiener filter, 260
windowing, 21
wireless network, 381

Z

z-scoring, 128
zebra finch, 307