Tampere University

TONI HEITTOLA

Computational Audio Content Analysis in Everyday Environments

Tampere University Dissertations 434

Tampere University Dissertations 434

TONI HEITTOLA

Computational Audio Content Analysis in Everyday Environments

ACADEMIC DISSERTATION
To be presented, with the permission of
the Faculty Council on Computing and Electrical Engineering
of Tampere University,
for public discussion in the auditorium TB109
of the Tietotalo building, Korkeakoulunkatu 1, Tampere,
on 18 June 2021, at 12 o’clock.

ACADEMIC DISSERTATION
Tampere University, Faculty of Information Technology and Communication Sciences
Finland

Responsible supervisor and Custos	Professor Tuomas Virtanen Tampere University Finland	
Pre-examiners	Associate Professor Romain Serizel Université de Lorraine France	Assistant Professor Mark Cartwright New Jersey Institute of Technology United States of America
Opponents	Associate Professor Romain Serizel Université de Lorraine France	Associate Professor Dan Stowell Tilburg University Netherlands

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

Cover design: Roihu Inc.

ISBN 978-952-03-2005-8 (print)
ISBN 978-952-03-2006-5 (pdf)
ISSN 2489-9860 (print)
ISSN 2490-0028 (pdf)
http://urn.fi/URN:ISBN:978-952-03-2006-5

PunaMusta Oy – Yliopistopaino
Joensuu 2021

PREFACE

This work has been carried out at the Department of Signal Processing, Tampere University of Technology, and Tampere University between 2009 and 2020. I wish to express my gratitude to my supervisor, Professor Tuomas Virtanen for giving me the opportunity to focus on such an interesting and novel research topic over an extensive period of time during the pivotal period of research in the field. This has allowed me to gain a deep understand of the topic and explore many aspects of the topic through research. His guidance during many research projects has been exceptional, and the high standards he has always set for the quality of the research has made me the researcher I am now. Furthermore, I wish to thank my former supervisor Dr. Anssi Klapuri for his supervision during my first years on this topic, and his valuable guidance during my first steps in academic research.

I would like to thank my co-authors Annamaria Mesaros, Antti Eronen, Emmanouil Benetos, Peter Foster, Mathieu Lagrange, and Mark D. Plumbley, for the work included in this thesis. Their contribution was important for this thesis and the collaboration always pushed research further. I would especially like to thank Annamaria Mesaros for a tight collaboration, always interesting discussions, and sharing the same passion in research and in life. I would also like to thank the pre-examiners of this thesis, Romain Serizel and Mark Cartwright, and the opponents of the public defense of this thesis, Romain Serizel, and Dan Stowell.

My great appreciation goes to all audio data collectors and annotators who have worked in the data collection campaigns for the audio datasets used in this thesis. Their work is extremely important for our research. I also thank the members of DCASE community for working together in advancing research in the field with great leaps in the recent years.

The Audio Research Group has provided a friendly working environment, and I am grateful for being part of it for all these years. I would like to thank all the past and present group members that I had the privilege of knowing and having insightful discussions about audio research and life in general, including, but not limited to Jouni Paulus, Mikko Parviainen, Pasi Pertilä, Julio Carabias Orti, Tom Barker, Joonas Nikunen, Aleksandr Diment, Shuyang Zhao, Emre Cakir, Sharath Adavanne, Paul Magron, Guangpu Huang, and Archontis Politis.

The funding and financial support received from Tampere Graduate School in Information Science and Engineering (TISE) and Nokia Foundation is gratefully acknowledged. This work was partly supported by the Academy of Finland (Finnish Centre of Excellence 2006–2011), and European Research Council under the ERC Grant Agreement 637422 EVERYSOUND (2015-2020). Support from Tampere University for finalizing the thesis manuscript is also gratefully acknowledged.

Lastly, I thank my mother for giving me the perfect tools for life and supporting my early interest in knowledge, science, and technology. I thank also my family, Annamaria for providing the beautiful constant factor for life, Milla and Salla for providing the delightful random elements, and our cats for providing the playful chaos.

Toni Heittola
Tampere, 2021

ABSTRACT

Our everyday environments are full of sounds that have a vital role in providing us information and allowing us to understand what is happening around us. Humans have formed strong associations between physical events in their environment and the sounds that these events produce. Such associations are described using textual labels, *sound events*, and they allow us to understand, recognize, and interpret the concepts behind sounds. Examples of such sound events are dog barking, person shouting or car passing by.

This thesis deals with computational methods for audio content analysis of everyday environments. Along with the increased usage of digital audio in our everyday life, automatic audio content analysis has become a more and more pursued ability. Content analysis enables an in-depth understanding of what was happening in the environment when the audio was captured, and this further facilitates applications that can accurately react to the events in the environment. The methods proposed in this thesis focus on sound event detection, the task of recognizing and temporally locating sound events within an audio signal, and include aspects related to development of methods dealing with a large set of sound classes, detection of multiple sounds, and evaluation of such methods.

The work presented in this thesis focuses on developing methods that allow the detection of multiple overlapping sound events and robust acoustic model training based on mixture audio containing overlapping sounds. Starting with an HMM-based approach for prominent sound event detection, the work advanced by extending it into polyphonic detection using multiple Viterbi iterations or sound source separation. These polyphonic sound event detection systems were based on a collection of generative classifiers to produce multiple labels for the same time instance, which doubled or in some cases tripled the detection performance. As an alternative approach, polyphonic detection was implemented using class-wise activity detectors in which the activity of each event class was detected independently and class-wise event sequences were merged to produce the polyphonic system output. The polyphonic detection increased applicability of the methods in everyday environments substantially.

For evaluation of methods, the work proposed a new metric for polyphonic sound event detection which takes into account the polyphony. The new metric, a segment-based F-score, provides rigorous definitions for the correct and erroneous detections, besides being more suitable for comparing polyphonic annotation and polyphonic system output than the previously used metrics and has since become one of the standard metrics in the research field.

Part of this thesis includes studying sound events as a constituent part of the acoustic scene based on contextual information provided by their co-occurrence. This information was used for both sound event detection and acoustic scene classification. In sound event detection, context information was used to identify the acoustic scene in order to narrow down the selection of possible sound event classes based on this information, which allowed use of context-dependent acoustic models and event priors. This approach provided moderate yet consistent performance increase across all tested acoustic scene types, and enabled the detection system to be easily expanded to new scenes. In acoustic scene classification, the scenes were identified based on the distinctive and scene-specific sound events detected, with performance comparable to traditional approaches, while the fusion of these two approaches showed a significant further increase in the performance. The thesis also includes significant contribution to the development of tools for open research in the field, such as standardized evaluation protocols, and release of open datasets, benchmark systems, and open-source tools.

TIIVISTELMÄ

Arjen ympäristömme ovat täynnä ääniä jotka auttavat ihmisiä ymmärtämään mitä heidän ympärillään tapahtuu, ja sitä kautta näillä äänillä on keskeinen rooli tiedon hankinnassa ympäristöstämme. Ihmiset muodostavat vahvoja assosiaatioita ympäristössä olevien fyysisten tapahtumien sekä niiden tuottamien äänten välille. Näitä assosiaatioita kuvataan tekstuaalisilla nimikkeillä, äänitapahtumilla, ja näiden assosiaatioiden avulla voimme ymmärtää, tunnistaa ja tulkita äänien takana olevat käsitteet. Esimerkkejä tällaisista äänitapahtumista ovat muun muassa koiran haukkuminen, ihmisen huutaminen tai auton ohi ajaminen.

Tämä väitöskirja käsittelee laskennallisia menetelmiä äänisisällön analyysiin jokapäiväisissä ympäristöissä. Lisääntyneen digitaalisen äänen käytön myötä automaattisesta äänisisällön analyysistä on tullut yhä tarpeellisempaa. Äänen sisältöanalyysi mahdollistaa syvällisen ymmärryksen siitä mitä ympäristössä tapahtuu hetkellä jolloin ääni tallennettiin, ja tämä puolestaan mahdollistaa sovelluksia jotka reagoivat tarkasti tapahtumiin ympäristössä. Väitöskirjassa ehdotetut menetelmät keskittyvät äänitapahtumien havaitsemiseen, laskennalliseen tehtävään jossa tavoitteena on tunnistaa äänitapahtuma sekä löytää ajanhetki jolloin se on aktiivinen äänisignaalissa. Väitöskirjatyö keskittyy kehittämään menetelmiä jotka pystyvät käsittelemään suurta joukkoa tunnistettavia ääniluokkia ja havaitsemaan useita ääniluokkia yhtä aikaa. Lisäksi työ paneutuu näiden menetelmien suorituskyvyn arviointiin.

Tässä väitöskirjassa esitelty työ keskittyy sellaisten menetelmien kehittämiseen jotka mahdollistavat useiden päällekkäisten äänitapahtuminen havaitsemisen sekä robustien akustisten mallien oppimisen äänisignaaleista jotka sisältävät päällekkäisiä ääniä. Työ lähtee liikkeelle Markovin piilomalli (HMM) pohjaisesta tekniikasta yhden hallitsevan äänitapahtuman havaitsemiseen kulloisenakin ajanhetkenä josta työ etenee polyfoniseen havaitsemiseen käyttäen joko useita Viterbi-iteraatioita tai käyttäen äänilähteiden erottelua esiprosessointimenetelmänä. Nämä polyfoniset äänitapahtumien havaitsemisjärjestelmät perustuvat joukkoon generatiivisia luokittelijoita jotka tuottavat useita ääniluokkanimikkeitä samalle ajan hetkelle. Tämä lähestymistapa kaksinkertaisti tai joissakin tapauksissa jopa kolminkertaisti äänitapahtumien havaitsemisen tarkkuuden. Vaihtoehtoisena lähestymistapana polyfoninen havaitseminen toteutettiin myös käyttämällä ääniluokkakohtaisia aktiivisuuden ilmaisimia.

Kunkin äänitapahtumaluokan aktiivisuus havaittiin itsenäisesti, ja yhdistämällä luokkakohtaiset tapahtumasarjat muodostettiin polyfoninen tunnistustulos. Polyfoninen havaitseminen lisäsi menetelmien soveltuvuutta jokapäivisissä ympäristöissä huomattavasti.

Menetelmien suorituskyvyn arviointiin väitöskirja ehdottaa uutta suorituskykymittaa joka ottaa huomioon äänitapahtumien polyfonian. Uusi suorituskykymitta, segmenttipohjainen F-score, tarjoaa tarkat määritelmät oikeille ja virheellisille havainnoille sekä soveltuu paremmin polyfonisten annotaatioiden ja järjestelmä ulostulojen vertailuun kuin aikaisemmin alalla käytetyt suorituskykymitat. Ehdotetusta mitasta on muodostunut sittemmin yksi vakiintuneista suorituskykymitoista tutkimusalalla.

Osa väitöskirjasta käsittelee äänitapahtumia osana äänimaisemaa käyttäen tapahtumien yhtäaikaista esiintyvyyttä kontekstuaalisena tietona. Tätä tietoa käytettiin sekä äänitapahtumien havaitsemisessa että äänimaisemien luokittelussa. Äänitapahtumien havaitsemisessa kontekstuaalista tietoa käytettiin rajaamaan mahdollisten äänitapahtumaluokkien joukko ensin tunnistamalla äänimaisemaluokka. Tämä lähestymistapa mahdollisti kontekstista riippuvien akustisten mallien sekä äänitapahtumien esiintyvyystodennäköisyyksien hyödyntämisen. Lähestymistapa lisäsi tasaisesti suorituskykyä kaikissa testatuissa äänimaisematyypeissä sekä mahdollisti järjestelmän toiminnan helpon laajentamisen uuden tyyppisiin äänimaisemiin. Äänimaisemien luokittelussa kontekstuaalista tietoa hyödynnettiin havaitsemalla maisemalle tyypillisiä äänitapahtumia. Tämä lähestymistapa saavutti saman tasoisen suorituskyvyn kuin perinteinen lähestymistapa, joka perustuu äänimaiseman yleiseen akustiseen sisältöön. Näiden kahden lähestymistavan yhdistäminen tuotti merkittävän suorituskyvyn kasvun. Väitöskirja sisältää merkittävän panoksen tutkimusalan avoimen tieteen työkalujen kehitykseen. Väitöskirjatyössä on luotu standardoituja protokollia äänitapahtumien havainnoinnin tarkkuuden arviointiin sekä julkaistu avoimia äänitietokantoja, avoimia vertailu-järjestelmiä ja avoimen lähdekoodin työkaluja.

CONTENTS

ABBREVIATIONS

AEER	Acoustic event error rate
ASA	Auditory scene analysis
ASC	Acoustic scene classification
AT	Audio tagging
BLSTM	Bi-directional long short-term memory
CASA	Computational auditory scene analysis
CLEAR	Classification of events, activities and relationships
CNN	Convolutional neural network
CQT	Constant-Q transform
CRNN	Convolutional recurrent neural network
DCASE	Detection and classification of acoustic scenes and events
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
DNN	Deep neural network
DWT	Discrete wavelet transform
EM	Expectation-Maximization algorithm
ER	Error rate
FNN	Feedforward neural network
GMM	Gaussian mixture model
HMM	Hidden Markov model
HOG	Histogram of oriented gradients
kNN	k-nearest neighbor
LBP	Local binary pattern
LSTM	Long short-term memory

MFCC	Mel-frequency cepstral coefficient
MIR	Music information retrieval
NMF	Non-negative matrix factorization
PCA	Principal Component Analysis
PDF	Probability density function
PLSA	Probabilistic latent semantic analysis
ReLU	Rectified linear unit
SED	Sound event detection
SELD	Sound event localization and detection
SNR	Signal-to-noise ratio
SPD	Subband power distribution
SVM	Support vector machine
TF-IDF	Term frequency-inverse document frequency
UBM	Universal background model

LIST OF INCLUDED PUBLICATIONS

This thesis consists of the following publications, preceded by an introduction to the research field and a summary of the publications. Parts of this thesis have been previously published and the original publications are reprinted, by permission, from the respective copyright holders. The publications are referred to in the text by notation [P1]–[P7].

P1 **A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen**, "Acoustic Event Detection in Real Life Recordings," in *Proceedings of 2010 European Signal Processing Conference*, Aalborg, Denmark), pp. 1267–1271, 2010.

P2 **T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen**, "Context-Dependent Sound Event Detection," in *EURASIP Journal on Audio, Speech and Music Processing*, Vol. 2013, No. 1, 13 pages, 2013.

P3 **T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen**, "Sound Event Detection in Multisource Environments Using Source Separation," in *Workshop on Machine Listening in Multisource Environments*, (Florence, Italy), pp. 36–40, 2011.

P4 **T. Heittola, A. Mesaros, T. Virtanen, and M. Gabbouj**, "Supervised Model Training for Overlapping Sound Events Based on Unsupervised Source Separation," in *Proceedings of the 35th International Conference on Acoustics, Speech, and Signal Processing*, (Vancouver, Canada), pp. 8677–8681, 2013.

P5 **T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen**, "Audio Context Recognition Using Audio Event Histograms," in *Proceedings of 2010 European Signal Processing Conference*, (Aalborg, Denmark), pp. 1272–1276, 2010.

P6 **A. Mesaros, T. Heittola, and T. Virtanen**, "TUT Database for Acoustic Scene Classification and Sound Event Detection," *In 24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, pp. 1128–1132, 2016.

P7 **A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley,** "Detection and Classification of Acoustic Scenes and Events: Outcome of the DCASE 2016 Challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(2):379–393, Feb 2018.

Author's Contributions to the Publications

Toni Heittola is the main author in all of the the included publications except for [P1], [P6] and [P7]. In publication [P2]–[P5] where he is the main author, he has done all the research, implementation and written the majority of the publication content.

In publication [P1], he participated in the writing and planning of the experiment, and implemented the proposed methods. In publication [P6], he participated in the planning of data collection, created the dataset from the collected data, implemented the reference systems and participated in the writing of the publication. In publication [P7], he acted as coordinator for acoustic scene classification task and sound event detection in the real-life audio task of DCASE Challenge 2016, contributed to the release of datasets, reference systems, and to the analysis of the submitted systems in these tasks. He participated in the writing process for the portions describing these challenge tasks in the publication.

1 INTRODUCTION

Acoustic environments surrounding us in our everyday life are full of sounds which provide us important information for understanding what is happening around us. Humans have formed tight associations between events happening around them and the sounds they produce. These associations can be represented as a textual label, to label the individual sound instances as *sound events*. This thesis deals with computational methods for the analysis of everyday environments. The core methods proposed in the thesis involve detecting large sets of sound events in real-life environments. In natural environments, sound events often appear simultaneously, increasing the complexity of acoustic modeling of sound events and making the detection difficult due to interfering sound sources. Furthermore, acoustic modeling cannot make strong assumptions about the sound or its structure since generally sound instances of the same sound event class can have large inter-class variability.

Along with the increased usage of digital audio in our everyday life, automatic audio content analysis has become a more and more pursued ability. Content analysis enables an in-depth understanding of what was happening in the environment when the audio was captured, and this further facilitates applications that can accurately react to the events in the environment. When work for this thesis started, there was very little prior work on computational audio content analysis directed to everyday environments. The research had been focused on tightly controlled indoor environments, such as office or meeting rooms, with a limited set of sound classes. Furthermore, the existing systems were able to detect only the most prominent sound event at each time instance. The use cases for such systems are rather limited, as most of our everyday environments are much more diverse than these works were focusing on, and being able to detect only the most prominent event limits the performance substantially.

The work presented in this thesis focuses on breaking out from the limitations of the previous approaches by developing methods allowing the detection of multiple overlapping sound events and enabling robust acoustic model training based on mixture audio containing overlapping sounds. To support the methods development, part of the thesis focuses on the development of protocols for evaluating and measuring sound event detection performance.

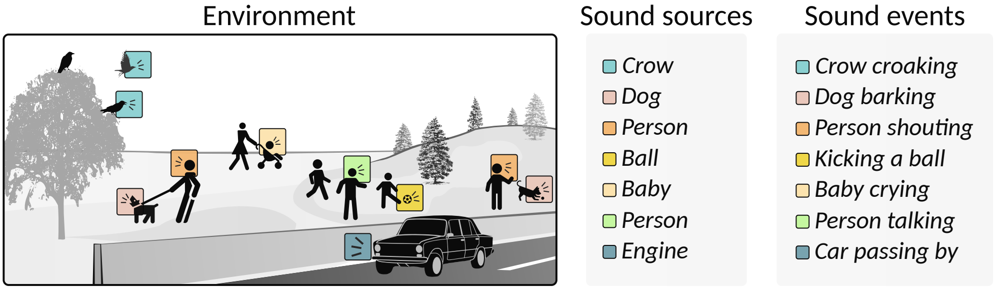


Figure 1.1 Examples of sound sources and corresponding sound events in an urban park acoustic scene.

1.1 Audio Content Analysis in Everyday Environments

Acoustic environments surrounding us in our daily life represent different *acoustic scenes* defined by physical and social situations. Examples of acoustic scenes include office, home, busy street, and urban park. *Everyday sound* is a term used to describe a naturally occurring non-speech and non-music sound that occurs in the acoustic environment [66]. The terms everyday sounds and environmental sounds are commonly used interchangeably in the literature. A *sound source* is an object or being that produces a sound through its own action or an action directed to it. A *sound event* is a textual label that people would use to describe this sound producing event, and these labels allow people to understand the concepts behind them and associate them with other known events. An example of an acoustic scene is shown in Figure 1.1, along with active sound sources and sound events associated with them.

Machine listening is a research field studying computational analysis and understanding of audio [195]. In this thesis, the term *audio content analysis* is used for approaches especially focusing on recognition of the sounds in the audio signal. Possible sounds in the audio signal include speech, music, and everyday sounds, however, this thesis focuses only on everyday sounds. The analysis system is said to do *sound event detection* (SED) if it provides a textual label and a start and end time to each sound event instance it recognizes. Sound event detection systems are categorized based on their ability to handle simultaneous sound events; if the system is able to output only a single sound event at time, often the most prominent, the system is said to do *monophonic* sound event detection, whereas the system capable of outputting multiple simultaneous sound events is said to do *polyphonic* sound event detection. In practice, current polyphonic state-of-the-art systems are not modeling or outputting multiple sound event instances from the same event class which are active at the same time, therefore the polyphony is defined in terms of distinct event classes. The content analysis systems which are only outputting sound class labels without temporal activity are said to do either *tagging* or *classification*, depending on whether the system is able to output multiple classes at a time or only a single class. These systems are closely related to detection, as they can be easily extended to output temporal activity with sufficient time resolution by applying classification in overlapping and consecutive short time segments. Even though SED is defined as detecting sound event instances, current modeling and algorithmic solutions treat the problem as audio tagging with fine temporal resolution and added temporal modeling of consecutive frames. This makes the distinction between detection and tagging dependent on the application. *Acoustic scene classification* (ASC) is a term used for systems classifying an entire recording into one of the predefined scene classes while in sound event detection or audio tagging the predefined classes are sound classes.

Applications

Audio content analysis is applied in a variety of applications to gain an understanding of the actions happening in the environment. It can be used, for example, in acoustic monitoring applications, in indexing and searching multimedia data, in analyzing human activity, and it is supporting research in neighboring research fields such as bioacoustics and robotics.

For monitoring applications, audio has many benefits over video capture: audio is often considered less intrusive than video, works equally well in all lighting conditions, does not require direct line-of-sight as video capture, and audio capture can cover large areas easily. Furthermore, computational requirements to handle audio in the analysis are far lower than for video, enabling the large-scale deployment of the monitoring applications. In surveillance and security applications, sound recognition and sound event detection can be used to monitor the environment for specific sounds, and once the sound has been detected trigger an alarm [33, 46]. Sounds of interest for these applications include, for example, glass breaking, sirens, gunshots, door slams, and screams. In healthcare applications, the same methods can be used, for example, to analyze cough patterns [61, 139] or to analyze epileptic seizures [5] over long periods to assist medical care personnel. In urban monitoring, audio content analysis methods can be used to identify sounds such as sirens, drills, and street music in urban environments and analyze their correlation to the noise complaints [12]. Sound recognition methods can be used also to assign noise level measurements to the actual sound sources in the environment, enabling more accurate noise measurements [100].

Many monitoring applications use small wireless devices, sensors, to capture audio. In case the sensor also has in-built audio content analysis capabilities, these sensors are referred to as *smart sound sensors*. These types of sensors are used when the overall system has to easily scale up from the computational resources and the wireless communication point of view. Instead of streaming captured audio or acoustic features extracted from it to the analysis service, the smart sensors are transmitting only information about the content of the captures audio, lowering the data transmission requirements substantially. Smart sensors are commonly used in smart homes [89] and smart city [12, 13] applications. In smart home applications, sensors are used to collect data from a home for security purposes or to assist home automation systems. Audio can be used to detect, for example, glass breaking or dog barking to trigger an alarm. In smart city applications, sensors are collecting a variety of sensory data to help manage resources in the cities, and audio can be included by using sound recognition approaches [100, 128].

Content-based analysis and search functionality is an important step on the way to fully utilize online services having large repositories of audio and video content. Audio content analysis approaches can be used in these services to enable content-based retrieval of multimedia recordings [80, 169, 202]. In addition to search functionality, content analysis methods can be used to automatically moderate the content.

Human activity is often the main source of sounds in everyday environments, and this is valuable information in many applications. Activities are usually broader concepts than sound events, for example, brewing coffee, cleaning, cooking, eating, or taking a shower. Audio content analysis can be used to identify and detect these activities, either by detecting individual sound events associated with the activity [28] or directly detecting activity concepts [131].

Audio content analysis can be used in other research fields to facilitate analysis of the environment or interaction with the environment. Bioacoustic research is nowadays utilizing more and more audio content analysis methods [167]. Methods can be used in wildlife population monitoring [165], animal species identification based on their vocalizations, and these analysis results can be further utilized in biodiversity assessment of the environments [53]. In monitoring applications for farming, audio content analysis can be used for assessing the animal stress levels [40, 95] and detecting symptoms of diseases [25]. In robotics, audio content analysis methods provide important information about the acoustic environment and actions in it. Social robots, such as home service robots can use sound event recognition to facilitate enhanced human-robot interaction [38, 81, 162].

1.2 Objectives and Scope of the Thesis

The main objectives of this thesis are:

- To develop methods for sound event detection with a large set of sound events and varying degree of polyphony. To solve how to handle overlapping sounds in the training stage, as well as in the detection stage.
- To develop an evaluation procedure for polyphonic sound event detection by defining appropriate metrics.

- To study sound events as constituent part of the acoustic scene.
- To develop tools for open research in the field: release open source reference systems and evaluation tools, and open datasets.

The main research questions studied in this thesis are:

Q1 How to implement a sound event detection system for a large set of sound events?

Q2 How to train acoustic models for sound events with audio containing overlapping sounds?

Q3 How to evaluate polyphonic sound event detection systems reliably?

Q4 How to distinguish between environments that have similar acoustic properties?

1.3 Main Results of the Thesis

The polyphonic sound event detection is at the core of this thesis, as it is an essential feature for a good performing sound event detection. As an application for sound event detection, detected sound events are used as mid-level representation in acoustic scene classification. The main contributions of the thesis are the following:

- A method for polyphonic sound event detection where overlapping event sequences are produced by using multiple restricted Viterbi passes. This addresses the first objective by answering the question Q1, and is presented in [P2].
- Methods to minimize the effect of interfering sounds during the acoustic model training by using audio material separated using unsupervised non-negative matrix factorization. This addresses the first objective by answering the question Q2, and is presented in [P3] and [P4].
- Sound event detection in everyday environments for a large set of sound events with varying degree of polyphony using context-dependent approach to dissect the detection problem into smaller and more easily manageable ones. This addresses the first objective by answering the question Q1, and is presented in [P1] and [P2].
- A new metric that accounts for polyphony, better suited for evaluation of polyphonic sound event detection than previously used metrics which were adopted from speaker diarization. This addresses the second and the fourth objective by answering the question Q3, and is presented in [P2] and [P3].
- Standardization of the evaluation procedure for polyphonic sound event detection through metrics and promotion of open science in the field by releasing open datasets and source code. This addresses the second and the fourth objective by answering the question Q3, and is presented in [P6] and [P7].

- Using sound events as a mid-level representation for acoustic scene classification. This addresses the third objective by answering the question Q4, and is presented in [P5].

The results and the contributions of each included publication are summarized in the following.

[P1] Acoustic event detection in real life recordings

The publication presents a system for monophonic sound event detection in recordings from everyday environments. The sound events are modeled using a network of hidden Markov models; model topology and size of individual sound event models are determined based on a study on isolated sound event classification. The publication is the first in the literature to evaluate sound event detection in a large-scale setting; 61 sound event classes are detected in 10 environments (over 15 hours of audio). The sound event detection system is capable of detecting a single most prominent sound event at a time. The proposed system was capable of recognizing almost one-third of the events, but the temporal positioning of the events is not correct for 84% of the time.

[P2] Context-Dependent Sound Event Detection

The publication introduces the concept of polyphonic sound event detection, where multiple simultaneous sound events are detected. Information about the acoustic scene class is incorporated into the system, and the benefits of such information are studied. The approach is motivated by human perception where context information is used to make more accurate sound event predictions and ruling out highly unlikely events given the context. The system introduced in [P1] is extended with an acoustic scene classification front-end and polyphonic detection is performed by using multiple restricted Viterbi passes to detect multiple event sequences. The proposed approach was found to improve detection performance substantially compared to the monophonic system proposed in [P1] or a context-independent system. By using the proposed context-dependent event detection scheme, the detection performance was almost doubled in comparison to the context-independent system.

[P3] Sound Event Detection in Multisource Environments Using Source Separation

The publication proposes a polyphonic sound event detection system where sound source separation is used as front-end to minimize the effect of interfering sounds. Incoming audio is pre-processed using unsupervised non-negative matrix factorization to separate audio into four audio streams representing a lower number of combinations of the physical sources than the original audio. A similar detection system to the one introduced in [P1] is applied separately on each of the four separated audio streams. The system allows detection of maximum four simultaneous sound events. The publication also proposes a new metric for evaluating event detection with various levels of polyphony: F-score calculated in non-overlapping segments. The proposed system showed a significant increase in event detection performance compared to the system proposed in [P1].

[P4] Supervised model training for overlapping sound events based on unsupervised source separation

The publication presents an extension to the system proposed in [P3] to train reliable acoustic sound event models by iteratively selecting the most appropriate training material from separated audio streams. Two approaches based on the expectation-maximization algorithm are proposed to select during the training the stream most likely to contain the target sound: one by selecting always the most likely stream, and another one by gradually eliminating the most unlikely streams from the training. Both proposed approaches were found to give a reasonable increase of 8 percentage units in the detection accuracy over [P3].

[P5] Audio context recognition using audio event histograms

The publication proposes acoustic scene classification based on representing each acoustic scene class using a histogram of sound events. In the training stage, each scene class is modeled with a histogram estimated from annotated training data. In the test stage, individual sound events are detected using the system presented in [P1], and a histogram of the sound event occurrences is built. The acoustic scene is recognized by calculating cosine distance between this histogram and event histograms from the training data, and the importance of different events in the histogram distance calculation is controlled by term frequency–inverse document frequency weighting. Event histogram based classification achieved 89% classification accuracy, and it further improved to 92% by combining histogram and conventional audio based recognition.

[P6] TUT database for acoustic scene classification and sound event detection

The publication introduces two open datasets to facilitate open research in the field: a first large scale dataset for acoustic scene classification, TUT Acoustic Scenes 2016 dataset of binaural recordings from 15 acoustic environments, and a first public dataset for sound event detection in real environments. For sound event detection, recordings from two environments were manually annotated with onset, offset and label of sound events, and the dataset was released as TUT Sound Events 2016. The publication presents the recording and annotations procedure for the datasets, the recommended cross-validation setup for system evaluation with these datasets, and a baseline system using mel frequency cepstral coefficients as features and Gaussian mixture models as a classifier.

[P7] Detection and classification of acoustic scenes and events: outcome of the DCASE 2016 challenge

The publication summarizes the second edition of the public evaluation campaign on detection and classification of acoustic scenes and events (DCASE 2016): introducing challenge tasks and their baseline systems, datasets used in the challenge, metrics used in the evaluation, and a thorough analysis of systems submitted to the challenge. The challenge included four tasks: acoustic scene classification, sound event detection in synthetic audio or in real-life audio, and domestic audio tagging. This edition of the challenge highlighted the emergence of deep learning in the field of content analysis of everyday environments, as most of the top-performing submissions used deep neural network based solutions.

Complementary material

In addition to the included publications, a large number of publications co-authored by the author of this thesis support and further develop the included studies. Parts of the thesis introduction are based on selected supplementary publications: a book chapter [72] and two journal publications [109, 113]. These publications present a more general overview of the gradual development in the field, rather than specific studies, and include the general machine learning approach for sound event detection [72], meta analysis of several approaches for sound event detection [109], and a comprehensive presentation of evaluation methodology for polyphonic sound event detection.

1.4 Organization of the Thesis

This thesis is organized as follows. Chapter 2 gives an overview of human perception of the everyday environments: how everyday sounds are identified and how they are categorized. The background information about the processing stages in audio content analysis system are presented in Chapter 3. Chapter 4 makes a complete presentation of sound event detection approaches proposed in this thesis and their evaluation results. The chapter also goes through evaluation procedures for sound event detection, introduces the metrics and datasets. Finally, Chapter 5 summarizes the contributions and discusses the future directions for content analysis in everyday environments.

2 SOUNDS IN EVERYDAY ENVIRONMENTS

Our everyday environments are naturally full of sounds. However, not all of them are considered equally relevant to the listener. During evolution, our auditory perception has evolved to capture meaningful sounds as this was necessary for finding food, avoiding hazardous situations, and communicating with other humans [197]. The sounds in our everyday environments can be grouped roughly into three perceptual groups: speech, music, and *everyday sounds* [7, 69]. Speech can be almost always considered to refer to sound which is produced by the human speech production system and having linguistic content. It is arguably the most important sound type in our everyday environments for its use in communication and social interaction. Music, on the other hand, is structured sound organized to transmit aesthetic intent. Everyday sounds, the third perceptual group, is the most diverse in terms of sound types and contains all the other sounds from our everyday environments.

The study of auditory perception has historically mainly focused on speech and music sounds under tightly controlled experimental environments, but in the last few decades, everyday sounds have been increasingly studied. The studies take an ecological approach to auditory perception, studying the auditory perception in natural environments and focusing on events creating the sound rather than specific psychoacoustics of the sound [55, 56]. Speech and music sounds have a strong temporal, spectral, and semantic structure on which the auditory perception can be based on. In contrast, everyday sounds do not have a predefined or recurring structure like speech and music, and thus audio containing everyday sounds is often referred to as *unstructured audio*. For everyday sounds, the meaning of the sound is commonly inferred directly based on the auditory properties of the sound (nomic mapping), whereas the perception of speech and music sounds relies more on arbitrary and learned associations (symbolic mapping) [54]. Sequences of everyday sounds do not follow any syntactic rules like speech or music sounds, although there are some short sequences of sounds that have a meaning [8]. The main properties of speech, music, and everyday sounds are collected in Table 2.1.

This chapter goes through the fundamentals of everyday sound perception and focuses particularly on the aspects applicable in the computational audio content analysis research. This knowledge can be used in various stages of development to

Table 2.1 Comparison of spectral, temporal, and semantic structure of speech, music and everyday sounds [66].

Speech	Music	Everyday sounds
	General characteristics	
produced by human speech system, analysis typically based on phonemes	produced by musical instruments, analysis typically based on notes	produced by any sound-producing events, analysis typically based on events
	Spectral structure	
mostly harmonic, some inharmonic parts	mostly harmonic, some inharmonic parts	unknown proportion of harmonic to inharmonic parts
	Temporal structure	
more steady-state than dynamic	mix of steady-state and transients, strong periodicity	unknown ratio between steady-state and dynamic, variable periodicity
	Semantic structure	
symbolic mapping, grammatical rules	symbolic mapping, music theory	nomic mapping, no structure or rules, some meaningful sequences exist

make informed design choices, whereas in the final system evaluation it provides insights on which sounds are meaningful in the context and which confusions are more acceptable than others. Furthermore, knowledge about the human categorization of everyday sounds and how sounds are organized in taxonomies can be used when designing and collecting audio datasets and creating reference annotations for audio content analysis research. For a comprehensive introduction to everyday sound perception see [65, 96].

2.1 Perception of Auditory Scenes

A sound is produced when an object vibrates and causes the air pressure to oscillate. The vibration is usually triggered by some physical action applied to the object. In the case of multiple simultaneously active sound sources, the air pressure variations caused by these sources are summed up and form an additive mixture signal. The term *auditory scene* is used when referring to complex auditory environments where sounds are overlapping in time and frequency.

Auditory System

The variations of the air pressure reaching the ear are converted into nerve impulses inside the ear and these impulses are analyzed in the auditory cortex of the brain. To create the nerve impulses, the sound is first converted into mechanical energy by the eardrum and then in the cochlea this mechanical energy is transformed into nerve impulses. The cochlea breaks sound into logarithmic frequency bands and each frequency band produces its own neural response, essentially producing a spectral decomposition of the sound [140].

Psychoacoustics, a research field combining acoustics and psychology, has established connections between the acoustic characteristics of the input signal and subjective properties of the sound perceived in the auditory system. The most common properties are *pitch*, *loudness*, and *timbre* of the sound. Pitch is related to the fundamental frequency of the sound, whereas loudness is related to the perceived intensity of the sound. Timbre is a multidimensional property of the sound related to the spectro-temporal content of the sound allowing sounds to be distinguished from each other. The main dimensions identified for timbre are related to the balance of energy in the spectrum (sharpness and brightness), the perception of amplitude modulation in the signal (fluctuation strength and roughness), and characteristics of sound start (onset). Timbre is an important sound property when identifying the sound sources.

The auditory processing stages in the human ear and the psychoacoustical studies on timbre perception have inspired the design of state-of-the-art acoustic features. These features are discussed in Section 3.3.2 and used in [P1]–[P7].

Auditory Scene Analysis

Auditory perception organizes acoustic stimuli from the auditory scene into *auditory objects* and identifies the corresponding *sound events* for them. The auditory object is a fundamental and stable unit of perception, acquired through grouping and segregation of spectro-temporal regularities in the auditory scene [15, 16]. An overview of this process is illustrated in Figure 2.1 with two overlapping sound sources.

A widely accepted theory for auditory perceptual organization, *auditory scene analysis* (ASA) [16], suggests that auditory perception organizes acoustic stimuli based on rules originating from Gestalt psychology. Auditory objects are perceived as sensory entities, which are formed following primitive grouping principles based on similarity, continuity, proximity, common fate, closure, and disjoint allocation. The similarity principle groups together components sharing perceptual properties (e.g pitch, loudness, or timbre). Continuity and common fate are related to the temporal coherence across or within the perceptual properties. The continuity principle

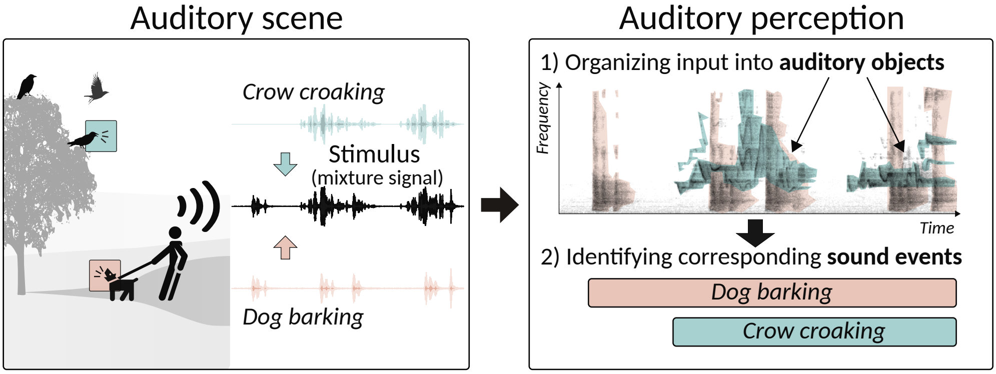


Figure 2.1 An example showing auditory perception in auditory scene with two overlapping sounds.

assumes sound to have only smooth variations in its perceptual properties across time; abrupt changes in these properties are considered as cues for grouping. The proximity principle groups together components close by either in frequency or time. The common fate principle looks into correlated changes in the perceptual properties, for example, grouping components having common onset or frequency modulation. Based on prior knowledge of the sound, the closure principle assumes sound to continue, even if there is another sound masking the original sound temporally, until there is perceptual evidence that the sound has stopped. Lastly, disjoint allocation refers to the principle of associating a component only into a single auditory object at a time. This primitive grouping works in a data-driven bottom-up manner. In addition, a schema-based grouping that works in a top-down manner is also proposed in ASA. The schema-based grouping utilizes the learned patterns and is commonly used with speech and music sounds. Both grouping types associate a group of sequential and overlapping components of sounds into an auditory object, and when these auditory objects are linked in time they form an auditory stream. This allows the listener to follow a particular sound source in the complex auditory scene, a feature traditionally called in the scientific literature as *cocktail party effect*.

Computational methods inspired by the auditory scene analysis and human auditory perception are studied under the research field called Computational Auditory Scene Analysis (CASA) [31, 168, 193]. These methods aim to derive properties of individual sound sources from a mixture signal, and approaches used are perceptually motivated. Most studies related to CASA deal with speech and music sounds, and they usually make strong assumptions about the characteristics of the input signal. In addition, the target sound source is often assumed to be in the foreground of the auditory scene, and the task is to separate it from the background. Computational methods targeting everyday sounds cannot make such assumptions about the input signal, as the sounds are diverse in characteristics and they can be either in the foreground or background in the auditory scene. Similarly to the auditory perceptual organization, sound source separation aims to decompose a mixture signal into individual sound sources, which can be then recognized with sound event detection methods. Sound source separation will be discussed in Section 3.3.1; the technique was applied in [P3] and [P4].

2.2 Perception of Everyday Sounds

Everyday sound perception, *everyday listening*, does not focus directly on the properties of sound itself. Instead, the focus is on the event which is producing the sound, and on the sound source, in order to understand what is happening in the surrounding environment. At the same time, everyday sounds are not listened to actively all the time; they are passively listened to until a sound of interest occurs, after which the auditory perception switches into active listening mode to identify the sound event. Everyday listening then segregates the perceived auditory scene into distinct sound sources and identifies corresponding sound events to these sound sources [56]. For example, when a car is driving on the road and passes the listener, first it catches the listener's attention and perception enters into active listening mode; after this, perception identifies the sound source (car) and the physical action causing the sound (driving), and associates the sound with the sound event "car passing by".

For everyday listening, identification of the sound event is essential to gain an understanding of the environment, while for the perception of speech or music the sound source is usually known already or identification has lower importance [56]. Sound *identification* can be seen as the cognitive act of sound *categorization*. Through categorization, humans are making sense of the environment by organizing it into meaningful categories, essentially grouping similar entities. This way, humans can handle the variability and complexity of everyday sounds and reduce the perceived complexity of the environment [65]. Categorization allows humans to hypothesize the event which produced the sound even if they have not heard the sound before.

Properties of Everyday Sounds

Humans can perceive properties of the sound-producing object as well as properties of the sound-producing action. In psychomechanics, a variety of experimental studies have focused on the perception of isolated physical properties of sound sources such as material [103], shape [92], and size [63], and parameters of sound-producing actions [97].

A deeper understanding of properties necessary for sound source identification can be acquired by degrading signals deliberately in various ways and studying test subjects' identification abilities with these signals. Experiments in [67] showed a reasonable ability to identify everyday sounds even when signals were filtered with low-pass, high-pass, or band-pass filters with varying filter cutoff or center frequencies, or when fine-grained spectral information was removed from the signals. On average, everyday sounds were found to contain more information in higher frequencies than speech. The most important frequency region for the identification was found to be 1.2-2.4 kHz, which is comparable to similar studies on speech signals. However, at the sound class level, there were large variations in the identification performance. When fine-grained spectral information was removed, the identification was mainly based on temporal information, and analysis showed that test subjects used envelope shape, periodicity, and consistency of temporal changes across frequency as cues for the identification. Again there was a large variation on the identification performance per sound class, half of the sound classes were identified correctly but some classes were not identified at all. It is worth noting that under similar conditions humans achieve near-perfect speech recognition performance [158]. These experiments highlight the diversity of everyday sounds and that identification of a wide range of different everyday sounds requires essentially full frequency information to work robustly.

Events in everyday environments do not occur in isolation, instead, they are usually happening in relation to other events and certain environments [133]. This contextual information enables humans to accurately identify acoustically similar sounding sounds. For example, in some conditions car engine noise and purring sound of a cat can be ambiguous, and contextual information helps disambiguate between them [8, 130].

The observations about the significance of full frequency information should be taken into account when designing an audio content analysis system for a diverse set of sound classes, and favor full-band audio. Contextual information can be used in automatic sound event detection systems to narrow down the selection of possible sound events and enable more robust detection similarly to human perception. These aspects will be discussed in Section 4.7; contextual information was used in [P2].

Categorization of Everyday Sounds

Humans categorize everyday sound events mostly based on the sound source (e.g. door slam) or action which generated the sound (e.g. squeaking), and only if the sound is unknown they fall back to describing the sound based on its acoustic characteristics [7, 186]. In addition, the location (e.g. shop) or context (e.g. cooking) in which the sound was heard and the person's emotional responses about the sound (e.g. pleasantness) have been found to affect the categorization [68]. Similarity has an important role in categorization, and various types of similarity have been found to be used in sound categorization: the similarity in acoustic properties (e.g timbre, duration), the similarity in the sound-producing events, and the similarity in the meaning attributed to the sound events [98].

Various categorization principles operate together, and they flexibly form varying types of categories related to the sound source, the action causing the sound, or context where the sound is heard. Early theories about the human categorization process proposed that categorization is based on the similarity between internal category representations and a new entity to be categorized. Depending on the situation, the category is represented either with a prototypical example that best represents the whole category [151], or with a set of examples for the category previously stored in the memory [160]. Categorization is done by inferring the category to a new entity from the most similar example. This results in a flexible categorization process with smooth boundaries between categories. These theories work in a bottom-up manner by processing low-level acoustic properties such as similarity, towards higher cognitive levels. Later theories extended this with a mixture of bottom-up and top-down processing where the data-driven bottom-up processing interacts with the hypothesis-driven top-down process that relies on expectations, prior knowledge, and contextual factors [79]. This type of processing is well suited for everyday sound perception: a person's prior knowledge about the categories and situational factors are used while doing the identification; however if there is no established prior knowledge about the categories, the identification is done in a data-driven manner by processing acoustic properties.

The knowledge about the categorization principles can be utilized when designing computational sound classification systems. The acoustic model in these systems acts as a set of internal category representations in human categorization, and classification is done by matching the unknown sound to this representation. Most computational sound classification systems can be seen to work in a bottom-up manner: acoustic features extracted from an unknown sound are used to infer the class label. However, some systems are also using top-down elements, for example, the contextual information to guide their classification process. Machine learning will be discussed in Section 3.4. The contextual information usage will be discussed in Section 4.7 and was used in [P2].

Organization of Everyday Sounds

Everyday sounds can be organized into *taxonomies* to assist the categorization process. In these taxonomies, the sounds are organized in a hierarchical structure according to sound sources [64], actions producing the sound [77], contexts where the sound can be heard [17], or combinations of these [56, 155]. The taxonomy proposed in [56] is based on the physical description of the sound production: at the highest level sounds are organized based on materials (vibrating solids, gasses, and liquids), under which sounds are organized based on actions producing the sound (e.g. impacts, explosion, or splash), and at the lower level sounds are organized based on interactions producing the sound (e.g. bouncing and waves). The taxonomy proposed in [155] for urban sounds starts with four categories (human, nature, mechanical, and music), and leaf nodes under them are related to either sound sources (e.g. laughter and wind) or sound-producing events (e.g. construction and engine passing). Everyday sounds can also be organized into *ontologies* in which unlike taxonomies, entities can have multiple relationships within the structure. The ontology proposed in [58], Audio Set Ontology, contains 632 sound events in a hierarchy with six categories at the top: human sounds, animal sounds, music, sounds of things, source-ambiguous sounds, and general environment sounds. Authors have published also a large dataset (4971 hours of audio) organized using this ontology.

The spontaneous creation of a textual label for a sound event is two-fold: if the listener recognizes the sound event, it is described by the event producing the sound and properties of this event; if the listener cannot identify the sound event, the description is based on acoustic properties of the signal [41]. In perceptual experiments, the process of selecting a label is often simplified by asking the listener to indicate the object (a noun) and action (a verb) causing the sound [7, 98].

Automatic sound classification systems can use the relationships in the hierarchical structures such as taxonomy or ontology in two ways: confusions could be allowed under the parent node during the learning process, and the parent-child relationships can be used in the classification stage by outputting a common parent node when encountering ambiguous sounds. Taxonomies or ontologies can be used to increase the consistency of the set of sound classes used in the audio content analysis system by enforcing the classes to be from the same level of the hierarchy. When annotating sound events for datasets for audio content analysis research, labels are often chosen based on the object-plus-action scheme, as will be discussed in Section 3.2.2 and was used in [P6] and [P7].

3 COMPUTATIONAL AUDIO CONTENT ANALYSIS

Natural sounds present in environmental audio have diverse acoustic characteristics due to a wide range of possible sound-producing mechanisms, and thus it is common that sounds categorized semantically into the same group have largely varying acoustic characteristics. Natural sounds such as animal vocalizations or footsteps have larger diversity than electronically produced sounds such as alarms and sirens. For general audio content analysis where a wide range of natural sounds is targeted, this poses major difficulty when developing the analysis system.

In a well-defined analysis case with a target sound category having a low-level of variation in its acoustic characteristics, one can manually develop a sound detector based on distinguishing characteristics such as sound activity on a specific frequency range (e.g. detecting fire alarms). However, in most practical use cases the analysis system is targeting a larger set of sounds having wider variations in their acoustic characteristics, making manual system development an impractical method. Computational analysis in this case calls for an extensive set of parameters, *acoustic features*, to be calculated from an audio signal and use of automatic methods such as *machine learning* [14, 39, 62, 127] to learn to differentiate the sound categories based on the calculated parameters. Most of the computational analysis systems presented in the literature use a *supervised learning* approach where manually labeled sound examples are used to teach the machine learning algorithm to differentiate unknown sounds into target sound categories. The system developer defines sound categories beforehand and collects a sufficient amount of labeled examples from each target sound category to develop and evaluate the system.

Labeling a sufficient amount of examples for supervised learning can sometimes be a laborious process. *Active learning* approaches can be used to minimize the amount of manual labeling work by letting the learning algorithm select examples for labeling. In this iterative process, the learner selects the best candidates for manual labeling, and these manually labeled examples are then used to improve the learner [207, 208, 209]. To avoid manual labeling altogether, one can use techniques such as *unsupervised learning* [39, p. 17] and *semi-supervised learning* [39, p. 18]. In unsupervised learning, groups of similar examples within the data are discovered and used as training examples for supervised learning. In semi-supervised learning, a small set of manually labeled examples is used to identify similar examples from a larger dataset with unlabeled examples, essentially increasing the amount of usable training material for supervised learning [37, 206]. This thesis concentrates on the supervised machine learning approach and how this approach can be applied to computational audio content analysis.

3.1 Content Analysis Systems

In principle, content analysis systems categorize the input audio into predefined sound categories, target sound classes. In the case of multiple target sound classes, the analysis systems can be divided into two groups; systems able to recognize only one sound class at a time and systems able to recognize multiple sound classes at the same time. In literature these are referred to *multi-class single-label* and *multi-class multi-label* approaches. The number of target sound classes in the analysis systems can vary widely based on application area from systems concentrating only on two classes (target sound class versus all the other sounds) to systems recognizing tens of classes. Often the number of classes is limited by the available development data, achievable accuracy, and possible computational requirements.

In case the analysis system outputs information about the temporal activity of the target sounds, the system is said to perform *detection*, whereas in case the analysis system only indicates whether the target sound is present within the analyzed signal, the system is said to perform *classification* or *tagging*, depending whether the system outputs one or multiple classes at the same time. Temporal information contains timestamps for when the sound instance starts, and for when it has ended. In literature, these timestamps are often referred to as *onset* and *offset* times.

From the application perspective, acoustic scene classification (ASC) is commonly seen as multi-class single-label classification and audio tagging (AT) as multi-class multi-label classification. In sound event detection (SED) applications, multi-class single-label classification is often referred to as *monophonic* sound event detection and multi-class multi-label classification as *polyphonic* sound event detection. These application types are illustrated in Figure 3.1.

The processing blocks of a typical content analysis system are presented in Figure 3.2. The input to the system is an audio signal which is captured with a microphone in real-time or read from a stored audio recording. The *audio processing* block performs *pre-processing* and *acoustic feature extraction*. Pre-processing is used to enhance characteristics of the audio signal which are essential for robust content analysis or separate target sounds from the background. In the acoustic feature extraction,

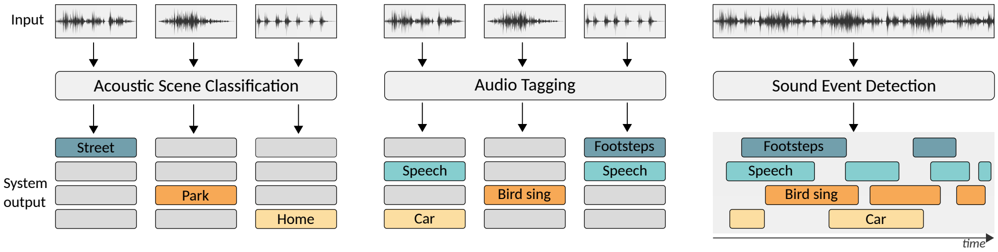


Figure 3.1 System input and output characteristics for three analysis systems: acoustic scene classification, audio tagging, and sound event detection.

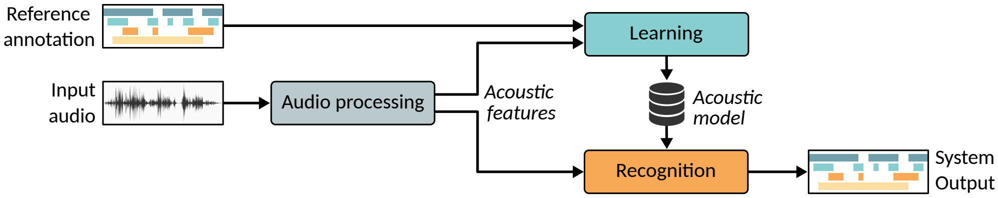


Figure 3.2 The basic structure of an audio content analysis system.

the signal is represented in a compact form by extracting information sufficient for classifying or detecting target sounds. This usually makes the subsequent data modeling in the learning stage easier with the limited amount of development examples available. Furthermore, the compact feature representation makes the data modeling computationally cheaper. During the system training stage, the acoustic features extracted in the audio processing block are used along with *reference annotations*. For the sound classification task, the reference annotations contain only information about the presence of target sound classes in each learning example, whereas for the sound event detection task onsets and offsets of these sounds may also be available. In the *learning* block, machine learning techniques are used to automatically learn the mapping between acoustic features and class labels defined in the reference annotations. In literature, the learned mapping is referred to as an *acoustic model*. In the *recognition* block, the previously learned acoustic models are used to predict class labels for new and previously unseen input audio signal. Depending on the application type, the system is doing either classification, tagging, or detection.

In the following sections, the data acquisition for the system development, and the techniques used in the processing blocks are described in detail. These sections are partly based on the introductory book chapter [72] about machine learning approaches for analysis of sound scenes and events published in [191].

3.2 Data Acquisition

Audio data and annotations describing the content of this audio data form together an *audio dataset* suitable for the development of the content analysis system. Data has a critical role in the development of systems based on machine learning techniques, as the performance level of such systems is strongly dependent on the quality and quantity of the data available during the development. Machine learning techniques used in the analysis rely on labeled data to learn parameters of the acoustic models and to evaluate the performance of these acoustic models for the given analysis problem. Acquiring suitable training and testing data is generally the most time-consuming stage of system development.

The target application defines the type of data needed for system development. Generally the aim is to collect acoustic material in conditions which are as close as possible to the envisioned use case of the analysis system. The collected material should contain a selection of representative examples of all sound classes targeted in the system. For example, a good material to develop a robust dog barking detector targeted for home surveillance applications use should contain material recorded in various environments related to home (indoor and outdoor, and varying room size, etc.), with a wide selection of dogs from different dog breeds (from small to large-sized) barking in as natural as possible setting with varying dynamics.

3.2.1 Audio

Most sound sources present in everyday environments have internal variations in their sound-producing mechanism which can be perceived as differences in the produced sound. This leads to high intra-class variability which has to be taken into account when collecting the audio material. The audio examples should be selected to provide good *coverage* and *variability* in sufficient *quantity* [111, p. 149]. Coverage ensures that the material contains examples from all relevant sound classes to the target application, whereas variability ensures that for each class there are examples captured in variable conditions with various sound-producing instances. Sufficient quantity of examples fulfilling the coverage and variability criteria enables the machine learning techniques to learn robust acoustic models that *generalize*, i.e., perform well on sound examples that were not encountered in the learning [62, p. 107]. More specifically, audio material for the development of an acoustic scene classification system should contain recordings from many locations belonging to the same scene classes, whereas material for the development of a sound event detection system should contain multiple sound instances from the same sound event class, recorded in variable conditions.

The variable conditions are characterized by the properties of acoustic environment (e.g. size and the shape of acoustic space, type of reflective surfaces), the capturing microphone and device, the relative placement of the sound source and the microphone, and interfering noise sources present. In realistic usage scenarios, all of the condition variations cannot be taken into account explicitly in the data collection. If the collected material represents only a subset of the possible conditions, this can cause a mismatch between the material used to develop the system and the material encountered in the real usage stage, which eventually leads to poor performance. Hence in the data collection it is advisable to make extra effort to minimize this data mismatch by capturing as representative set of data as possible, under all identified conditions. When material captured under variable conditions is used in the learning stage, the system is said to use a *multi-condition training* approach [99, p. 116]. Collected audio data can be diversified for the multi-condition training by adding artificially different impulse responses to it in the training stage [210], since many of the variable conditions are reflected in the *impulse response* which characterizes the overall acoustic characteristics of the captured audio signal [99, p. 206]. This approach requires obtaining recordings of the target source with as little external effects as possible and then convolving the audio signals with measured impulse responses from various real acoustic environments. The room impulse responses can be generated also with room simulation techniques [213, p. 191].

Interfering noise sources can be handled similarly to acoustic conditions. In usage cases where potential noise sources are known and stationary, the data can be easily collected under similar conditions. However, in cases where the types of noise sources are varying or the relative position of the noise source with respect to the capturing microphone varies, data collection under all matched conditions is impractical. Depending on the level of variability, one can be still successful by collecting material as diversely as possible and using a multi-condition training approach. Another feasible approach is to obtain recordings of target sound sources without any interfering noise and recordings with the noise sources alone, and simulate noisy signals by artificially mixing these with various signal-to-noise ratios (SNR) [109]. The artificial mixing approach can potentially produce larger quantities of relevant training material than a direct recording approach. On the other hand, the diversity of the available recordings influences and limits the variability of the artificially produced material.

3.2.2 Annotations

Supervised machine learning approaches require labeled sound examples, i.e., audio data with *reference annotations*. In the annotation procedure, portions of the acoustic signal containing target sound categories are indicated and stored in some machine-readable format. Manual annotation is done through audition, having per-

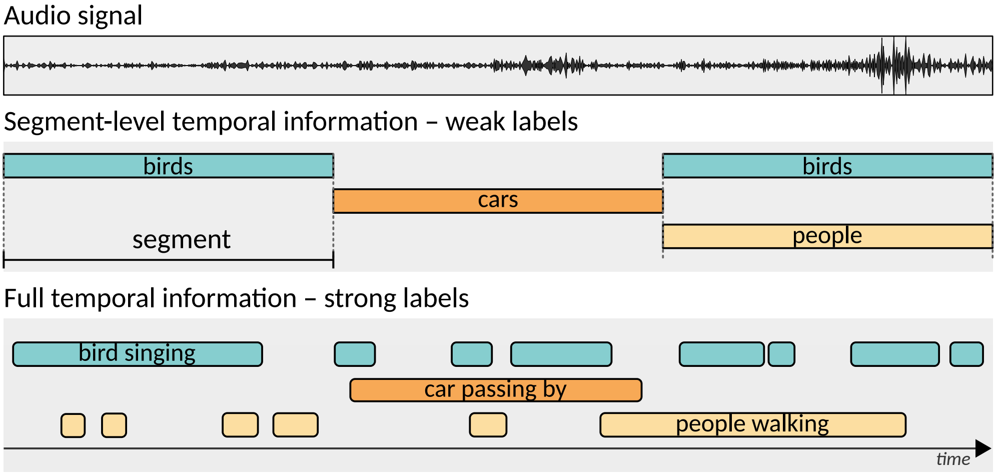

Audio signal
Segment-level temporal information – weak labels
birds
cars
birds
segment
people
Full temporal information – strong labels
bird singing
car passing by
people walking
time

Figure 3.3 Annotation with segment-level temporal information and with detailed temporal information.

sons carefully listening to the audio and indicating the activity of each class; because of this, the manual annotation process is one of the most time-consuming stages of the system development. Before the data collection process, the sound categories are selected based on the target application, and the textual labels assigned to these categories are defined to guide the selection of recording locations and situations. Selected textual labels should be representative and non-ambiguous, i.e., a label should allow understanding the sound properties based on the label alone in an explicit way [111, p. 152].

Audio content can be annotated at a fixed temporal grid, by annotating sound activity inside equal-sized and non-overlapping segments [50], or with detailed temporal information, by annotating the exact start and end times for the sound activity [P1], [P2] and [P6]. The annotations are *strong* when they have start and end times for the sound activity and *weak* when the temporal information is approximated at a coarse level (up to signal length). The most complex form of annotation for environmental audio is *polyphonic* sound event annotation, where multiple, overlapping sound events are annotated with strong labels [P6]. The different annotation types are illustrated in Figure 3.3. Depending on the content analysis application type, reference annotations have different requirements as listed in Table 3.1.

Audio material for acoustic scene classification is often captured in a fixed position to ensure that the scene category stays the same throughout the recording [P6] [123]. This simplifies the annotation process, as the category labels can be assigned for full signals or very long time-segments in it. Category labels should be clearly defined to minimize the subjectivity of the label selection in the annotation process. Examples of scene labels are *busy street*, *office*, and *traveling by bus*.

Table 3.1 Annotation requirements for three main content analysis types.

	Annotation unit		Temporal information	
Analysis type	Size	Overlap	Type	Typical resolution
Classification (ASC)	fixed	no	weak	≥ 1 s
Tagging (AT)	fixed	yes	weak	≥ 1 s
Detection (SED)	varying	yes	strong	≤ 1 s

Labeling sound events is a highly subjective process where perception and personal life experience of the annotator have an important role [69]. Subjectivity can be controlled to some extent by defining the textual labels for the sound categories before the annotation process and forcing the label selection among these pre-defined labels. This is advisable in applications where the number of target sound categories is low and they are well-defined. In research where the aim is to recognize all sounds in the acoustic scene, or the target application is not defined before the data collection, the labels cannot be defined before actually annotating the audio material. In these cases, the most advisable approach is to allow free label selection during the annotation, i.e., each sound instance will be annotated with a descriptive and possibly a new label, based on the annotator‘s opinion; afterwards labels describing the same sound category can be manually grouped after all material is annotated [P1], [P2], [P6], and [48]. The label post-processing stage is essential to make the material usable for supervised machine learning, as freely selected labels often contain typos, different wording (*people talking* versus *people speaking*, or synonyms (*car* versus *automobile*) for sounds clearly belonging to the same category.

Annotating sound events with full temporal information requires marking the time instance when the sound event is first perceived, and the time instance when the sound event is not anymore perceived. This temporal segmentation process will introduce a varying level of subjectivity to the annotations, depending on the sound event type [111, p. 157]. For sound events which have rapidly increasing and decreasing amplitude envelope, such as car horn, onsets and offsets can be pinpointed reliably with acceptable time resolution (e.g. 100 ms). On the other hand, for sound events which have slowly increasing and decreasing amplitude envelope, onsets and offsets can be sometimes very difficult to pinpoint reliably, especially when interfering background noise is present in the acoustic signal. Two such examples are illustrated in Figure 3.4.

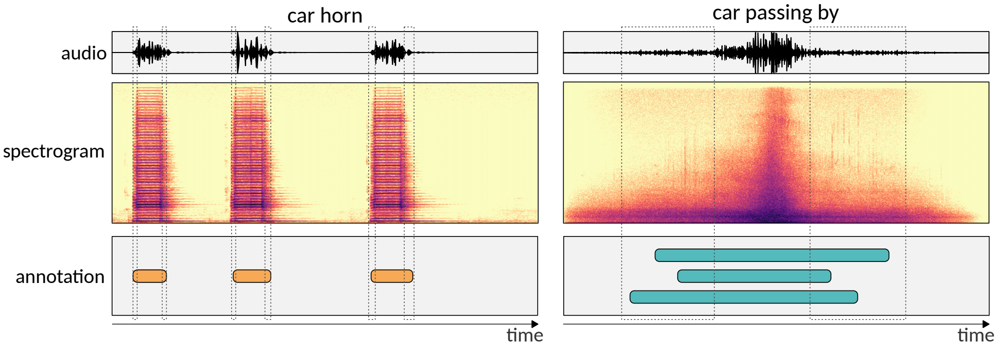


Figure 3.4 Annotating onset and offset of different sounds: boundaries of the sound event are not always obvious.

3.2.3 Datasets

Once the audio material is packed together with annotations, it forms an *audio dataset* usable for the content analysis development. It is useful if the dataset has additional metadata describing the recording equipment (e.g. microphone model, capturing device), recording time and location (e.g. address, GPS coordinates), and properties of the acoustic environment during recording (e.g. weather conditions, room size when indoor). This metadata has an important role in the creation of the cross-validation setup for the development when one creates balanced training, testing, and validation sets with respect to various properties of the data. For example, one should take extra care not to include recordings from the same exact location into training and testing sets, as this will potentially lead to over-optimistic performance estimates. Another example of the usage of the metadata is the creation of a cross-validation setup such that all sets contain a representative selection of recordings from different weather conditions.

For published datasets, it is good to follow a consistent file naming convention for a clear correspondence between audio recordings, related annotations, and metadata, to use easily accessible machine-readable file formats, and to include a cross-validation setup. A cross-validation setup supports a direct comparison of studies using the dataset, which is important for its usability and for reproducible science. More information about reference datasets for sound event detection can be found in Section 4.2

3.3 Audio Processing

In the audio processing stage of the content analysis system (see Figure 3.2), the audio signal is prepared and processed for the subsequent machine learning stage. The audio processing stage consists of *pre-processing* and *acoustic feature extraction*. In pre-processing, the audio signal is processed to reduce the effects of interfering noises or to emphasize the target sounds. In acoustic feature extraction, the audio signal is transformed into a compact representation suitable for machine learning algorithms.

3.3.1 Pre-processing

The aim of pre-processing is to enhance the characteristics of the audio signal that are essential for robust content analysis. The requirements for this processing block depend on the characteristics of the acoustic environment and the target sound categories, as well as the type of acoustic feature extraction and machine learning methods used. Pre-processing is generally applied to the audio signal before acoustic feature extraction, and prior knowledge about the usage environment and the distinctive characteristics of the target sound categories is utilized when designing or selecting the pre-processing algorithm. For example, if stationary noise is present in the operation environment, noise suppression techniques can be used to reduce the interference of noise to the analysis [157].

Everyday environments usually have multiple overlapping sound events active at the same time. The recognition of overlapping sounds can be addressed at different stages of the analysis system: at the signal pre-processing stage by using sound source separation [P4], at acoustic modeling level by modeling all sound combinations [19, 170], at detection level by using multiple iterative detection passes [P2]. Recently introduced deep neural network based approaches use large amounts of data to learn and recognize sounds regardless of the interference introduced by the overlapping sounds at the acoustic model level [24].

Sound Source Separation

Audio captured in our everyday environments consists of sounds produced by various sound sources having distinctive structure in time and frequency. Sound sources can be considered to correspond to sound events in the acoustic scene, sometimes multiple sound sources belonging to the same event. The aim of sound source separation is ideally to decompose a given audio signal, *mixture signal*, with multiple simultaneous active sound sources into individual sound sources.

One commonly used method for sound source separation is based on non-negative matrix factorization (NMF) [188]. NMF models the structure of the sound by representing the spectra of the mixture signal as a sum of *components*, each having a fixed magnitude spectrum and a time-varying gain. The assumption in NMF-based source separation is that each sound source has a characteristic spectral structure that differs from other sources present in the mixture signal, and ideally each source can

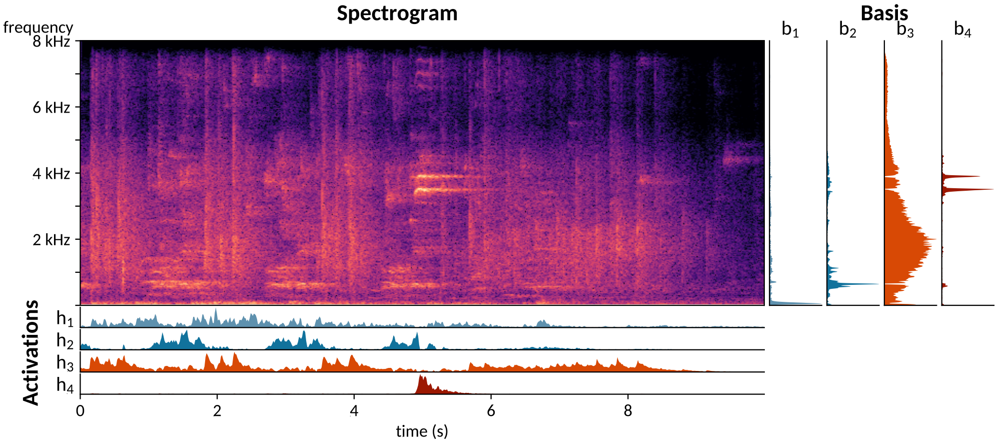


Figure 3.5 Sound source separation with NMF into four components applied to a recording captured in a basketball game.

be modeled using a distinct set of fixed magnitude spectra. In the signal model, the magnitude spectrum vector $\boldsymbol{x}_t$ in frame t is defined as a linear combination of basis spectra $\boldsymbol{b}_k$ (fixed spectrum) and corresponding $h_{k,t}$ activation coefficients (gain). This can be expressed as:

$$\boldsymbol{x}_t \approx \sum_{k=1}^{K} \boldsymbol{b}_k h_{k,t} \tag{3.1}$$

where K is the number of components, and $h_{k,t}$ is the activation coefficient of the kth basis spectrum in frame t, for $t = 1...T$ and T being the number of frames. One basis spectrum and its activation coefficients are referred to as a component. Often NMF is used in an unsupervised manner without any prior knowledge of which components represent the given sound source. Ideally sound sources in the mixture signal become represented as a sum of one or more components, however, it is possible that the resulting components contain parts from multiple sound sources. This is considered as learning-free sound separation, and it gives a good separation performance in cases where the characteristics of the sound sources are distinctive. Figure 3.5 shows the spectrogram of an audio signal captured during a basketball game and the results of factorization into four components. In this example, the first component captures the squeaking sounds of basketball players' shoes and residual audience sounds, the second component captures shouting from the audience, the third component captures applause, and the fourth component captures the whistle sound of the basketball referee.

The component-wise audio streams can be reconstructed by generating a time-frequency mask $\boldsymbol{w}_k$ from basis spectra and activation coefficients, and filtering the mixture signal with it. The time-frequency soft mask for component j is defined as

$$\boldsymbol{w}_j = \frac{\boldsymbol{b}_j \boldsymbol{h}_{j,t}}{\sum_{k=1}^{K} \boldsymbol{b}_k \boldsymbol{h}_{k,t}}. \tag{3.2}$$

The mask can be considered as a time-varying Wiener filter which separates the signal into a stream containing approximately homogeneous spectral content that differs significantly from the other streams. The outputted streams do not represent individual sound sources, but they are a combination of the sources present in the original mixture signal.

3.3.2 Acoustic feature extraction

The main purpose of feature extraction is to transform the acoustic signal into a compact numerical representation of the content in a way that is relevant to machine learning and maximizes the recognition performance of the sound analysis. Important information for the content analysis of audio signals is mainly contained in the relative distribution of energy in frequency. For this reason, regularly used acoustic features in audio content analysis are based on the *time-frequency representation* of the signal. The discrete Fourier transform (DFT) is the most commonly used transformation for audio signals. It represents the signal using sinusoidal base functions, each being defined by magnitude and phase [134]. Other transformations used for audio signals include constant-Q transform (CQT) [18] and discrete wavelet transform (DWT) [181].

Processing stages

Audio signals can generally be assumed to be non-stationary because of their rapidly changing signal statistics (e.g. magnitudes of the frequency components). This requires acoustic feature extraction in short time segments, *analysis frames*, which contain signal in a quasi-stationary state. The basic stages in acoustic feature extraction are *frame blocking*, *windowing*, *spectrum* calculation, and subsequent analysis, as illustrated in Figure 3.6.

In the frame blocking stage the audio signal is split into fixed-length analysis frames, which are shifted with a fixed time step (*feature hop length*). When using Fourier transform, the length of the analysis frame is related to frequency resolution: longer frames will give better frequency resolution than shorter frames, but at the same time the temporal resolution of the analysis is lower with longer frames. Usually for environmental audio analysis, the frame length is set between 20 and 100 ms with 25 - 50% overlapping frames. Sound event detection systems use shorter analysis frames

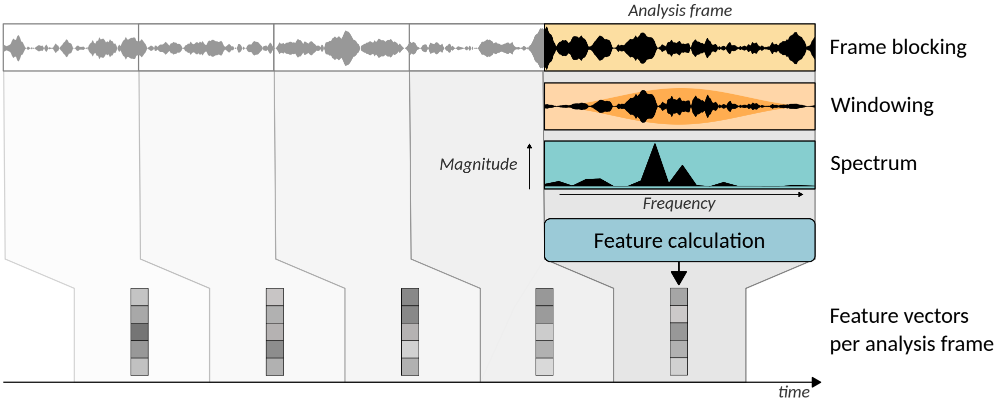


Figure 3.6 The processing pipeline of acoustic feature extraction.

(e.g. 20 ms in [P1]–[P3]) than acoustic scene classification systems, as spectral characteristics of sound events are changing more rapidly than the general characteristics of acoustic scenes, and good time resolution is required for detecting event onsets and offsets accurately. In order to avoid abrupt changes at the frame boundaries causing distortions in the spectrum, the analysis frames are smoothed with a *windowing function* such as Hamming or Hann function. The windowed analysis frames are transformed into a spectrum, forming a time-frequency representation, and acoustic features are extracted from it.

Until recently, the most common approach to develop acoustic features have been carefully engineering features from the time-frequency representation and using expert knowledge about acoustics, sound perception, sound classes, and their differences while developing features. These types of features are often called *hand-crafted features*. Recently, automatic *feature learning* techniques have also been used with increased dataset sizes [21, 153]. These techniques produce high-level feature representations given the data and specified task, and have shown impressive performance compared to hand-crafted features. The main advantage of feature learning over feature engineering is that no specific knowledge about the target task is required. This thesis only discusses hand-crafted acoustic features, and the most common hand-crafted features extracted from the spectrum, mel-band energies and decorrelated mel-band energies called mel-frequency cepstral coefficients (MFCCs) [34]. These features are illustrated in Figure 3.7 for an audio example.

Mel-band energies

Mel-band energies are a perceptually motivated representation based on mel-scaled frequency bands. The aim of the mel-scale is to mimic the non-linearity of human auditory perception, by having narrower bands at lower frequencies than at higher

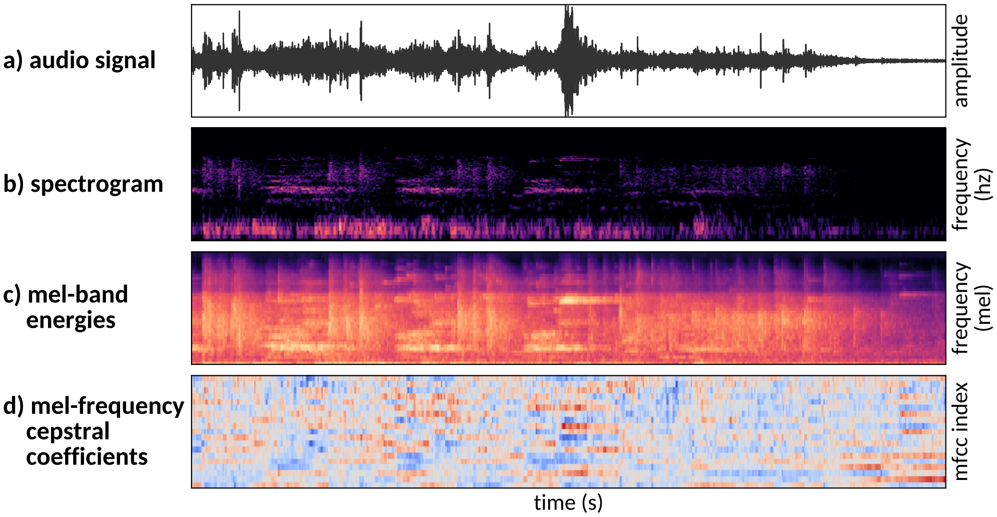


Figure 3.7 Acoustic feature representations.

frequencies. The scale has been created through listening experiments, having listeners listen to two alternating sinusoids and adjusting one of them to have a perceived pitch half to the other one [161]. The resulting mel-scale is approximately linear up to 1 kHz, after which it is approximately logarithmic. The relation between mel and linear frequency can be approximated as [132, p. 128]

$$F_{mel} = 2595 log_{10}(1 + \frac{F_{Hz}}{700}). \tag{3.3}$$

A mel-scale *filterbank* consists of overlapping band-pass filters (typically 20, 40, 64, or 128 filters in total) having triangular frequency response and with filters' center frequencies linearly spaced on the mel scale. Figure 3.8 illustrates the process of constructing such a filterbank; the relation between center frequencies of the filters in hertz and mels is shown in the top panel, and the triangular filters are shown in the bottom panel. Many alternative filterbank implementations have been proposed in the literature throughout the years, mostly varying in how the nonlinear pitch perception of humans is approximated in the filterbank design [34, 159, 204].

The mel-scale filterbank is applied on the spectrum (either magnitude or power spectrum) to obtain energy per *mel-band*. The resulting representation is called a *mel spectrogram*. Following humans' logarithmic perception of sound loudness, the dynamic range of the energy values per band is compressed by taking the logarithm of these values. The resulting features are called *log mel energies* in the literature. This acoustic representation retains the coarse shape of the spectrum, while the fine structure related to the harmonic structure of the signal is smoothed out. This

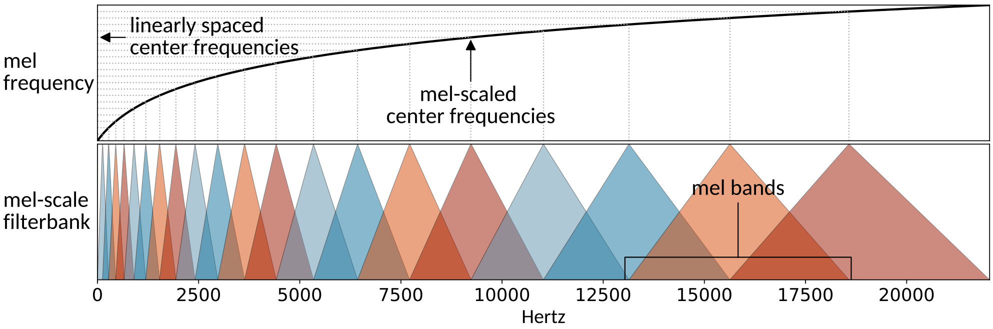


Figure 3.8 Mel-scaling (top panel) and mel-scale filterbank with 20 triangular filters (bottom panel).

is beneficial because the identity of everyday sounds is not determined based on the exact perceived pitch, and some of the sources do not even have any harmonic structure. The process is shown in Figure 3.7; panel b shows the spectrogram of the signal, and panel c shows corresponding mel-band energies.

The mel-scale filterbank was originally designed for speech analysis, and to be used as a processing block for MFCCs features. Later, as MFCCs were shown to perform robustly in more general sound classification tasks such as speaker recognition [148], music genre recognition [180], and musical instrument classification [71], they gained popularity also as a standard feature for acoustic scene classification and sound event detection tasks. MFCCs are calculated by decorrelating the outputs of the mel-scale filterbank with a linear transform, the discrete cosine transform (DCT) to allow efficient data modeling in Gaussian mixture models and hidden Markov models by enabling the usage of a diagonal covariance matrix in Gaussian distributions. Along with the emergence of deep learning based approaches the decorrelation step has been dropped out because modern deep learning techniques can efficiently take advantage of correlated information in the data during the learning process. Currently, log mel energies are the most commonly used acoustic features in deep learning based approaches for the content analysis of environmental audio [24, 109, 123].

Mel-frequency cepstral coefficients

Mel-frequency cepstral coefficients (MFCC) represent the output of the mel-scale filterbank in the *cepstral domain*. MFCCs are obtained from the previously described log mel energies by computing type-II discrete cosine transform (DCT). An example of MFCCs is shown in Figure 3.7, panel c.

The role of this added processing step is two-fold. Firstly, the DCT is used to decorrelate feature values, since the filterbank outputs are heavily correlated due to neighboring filters overlapping in frequency. Decorrelated values enable usage of a diagonal covariance matrix in Gaussian distribution based acoustic models.

Secondly, by keeping only the first few values of DCT (coefficients), the spectral representation is smoothed and the dimension of the feature vector is reduced. The first coefficients contain information about the overall shape of the spectrum, while higher coefficients contain information about the fine structure of the spectrum. The amount of coefficients retained for analysis varies between 12-20 depending on the target application requirements; in speech recognition 8-12 coefficients are sufficient to represent the coarse shape of the spectrum; in musical instrument recognition usually a higher number of coefficients (e.g. 20) are used to capture fine details of the spectrum [71], while in environmental audio usually 16-20 coefficients are used [P1]–[P6]. The first MFCC (zeroth coefficient) is related to signal energy (log energy), and depending on the application this information is either retained or omitted from the feature vector. Signal energy is closely related to acoustic scene class, e.g. park is quieter than a street with cars, and because of this, the signal energy information is retained in acoustic scene classification applications. Sound events can be regarded as having the same source regardless of loudness, and thus the zeroth coefficient is often omitted from the feature vector in sound event detection applications.

Dynamic features

The audio is a time-variant signal and one of the main characteristics for sound identification is its dynamic change over time. However, many acoustic features such as mel energies and MFCCs, estimate only the instantaneous spectral shape. The temporal evolution of the acoustic features can be dealt at the feature level by adding dynamic features to the final feature vector [P1]–[P6] or by modeling the temporal aspect in the acoustic modeling stage (e.g. using recurrent layers in neural networks) [24].

To incorporate the temporal evolution of the features into the acoustic feature vector, one can use estimates of the local time derivatives of the features by approximating with a first-order orthogonal polynomial fit [51] as

$$\Delta c(i,u) = \frac{\sum_{k=-K}^{K} k \cdot c(i, u+k)}{\sum_{k=-K}^{K} k^2} \tag{3.4}$$

where $c(i,u)$ denotes the i^{th} feature value in a time frame u [145, pp. 116-117]. Computation is performed over $(2K+1)$ frames, and K is typically set to either three or four. The resulting dynamic features are commonly referred to as *delta* features, opposite to the *static* features. In addition to the delta features, the second derivatives can be computed by applying the same polynomial fit to the already computed delta features, resulting in a parabolic fit. These features are referred to as delta-delta or *acceleration* features. Delta and acceleration features are used together with static

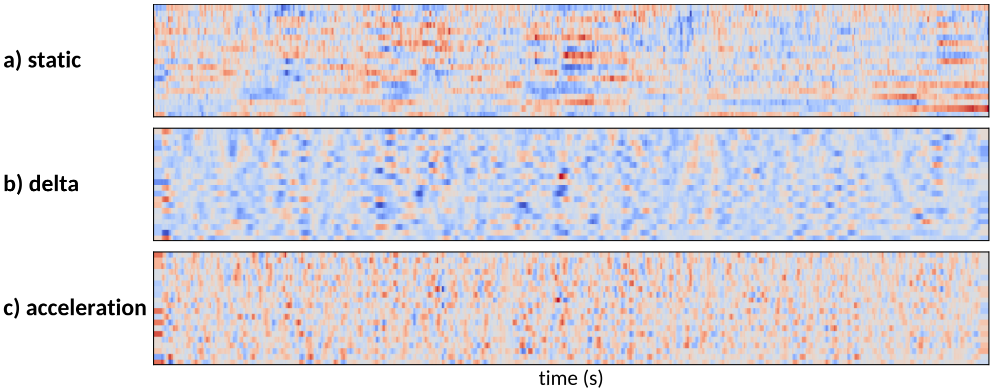


Figure 3.9 Static and dynamic feature representations.

features to integrate the dynamic aspect of the spectrum to the feature set [P1]–[P6]. An example of the static and dynamic features is shown in Figure 3.9.

Another method to incorporate temporal information into the feature vector is to concatenate consecutive frames within a window into a single vector, a supervector, [23, 59, 109]. The idea is to provide contextual information by stacking together N_{win} frames before and after the current frame. The length of the newly constructed feature vector is defined as $(2 \times N_{win} + 1) \times N_{feat}$, where N_{feat} denotes the length of the original feature vector. This method is often used together with fully connected feed-forward neural networks.

Other features

In addition to the previously discussed features, numerous other features derived from the time-frequency representation of a signal have been proposed for computational audio content analysis.

Low-level features describing specific aspects of the spectral shape of a signal are traditionally used as part of a larger feature set together with MFCCs as they are not powerful enough to be used alone. Common low-level features that describe spectral shape include signal energy, spectral envelope, spectral moments (e.g. spectral centroid and flatness), spectral slope, spectral roll-off, spectral flux, and spectral irregularity [44, 57].

Spectral descriptors adapted from image processing have shown competitive performance compared with traditional features, especially in the case of acoustic scene classification. These descriptors are treating the spectrogram as an image and use techniques adapted from computer vision to characterize the shape, texture, and evolution of the content in it. To detect different shapes in the spectrogram, one can use a histogram of oriented gradients (HOG) features [147], and to characterize textures in the spectrogram one can use local binary pattern (LBP) features [10, 85]. Subband power distribution (SPD) transforms the spectrogram into representation characterizing the spectral power distribution over time at frequency subbands [36].

Automatic representation and *feature learning* techniques have become recently increasingly popular for acoustic scene classification and sound event detection facilitated by the availability of larger high-quality datasets. Sounds occurring in everyday environments have substantial diversity and this results in a widely varying set of time-frequency structures, however, only a subset of the information in the spectrogram is relevant for actual classification. The aim of feature learning is to learn a representation that reflects this relevant information, and is accomplished using different techniques, including deep learning [76]. Features created through this process are commonly referred to as *embeddings*. The input for the feature learning network can be some established time-frequency representation or even raw acoustic waveform [84].

3.4 Supervised Learning and Recognition

Machine learning techniques used in audio content analysis rely on data to learn the parameters of the acoustic models. In a *supervised learning* approach, manually labeled sound examples are used to teach a *classifier* the mapping between the extracted acoustic features and given sound categories. The learned acoustic model is then used to assign category labels to acoustic features of the test data. The difficulty of the classification task depends on the inter-class and intra-class variability of sound categories. When doing sound classification or detection of everyday sounds, the used learning algorithm has to cope with overlapping sounds and take into account the temporal structure of sounds. Depending on the target task, the classification can be formulated as a multi-class single-label or a multi-class multi-label problem. In the multi-class single-label problem, only one category out of many possible categories is active at a given time instance, whereas in the multi-class multi-label problem multiple categories can be active simultaneously at a given time instance.

3.4.1 Learning

In the learning stage, the aim is to learn an optimal model to classify sound examples into one of the predefined sound categories in a given feature space. The learning examples are pairs of inputs to the system, acoustic features $\mathbf{x}_t$ extracted in time instances $t = 1, 2, ...T$, and desired target outputs $\mathbf{y}_t$ for those particular inputs. The target output contains the information about the sound class assigned to the input data, one of possible C classes, $c_i, i = 1, 2, ...C$. An overview of the learning process

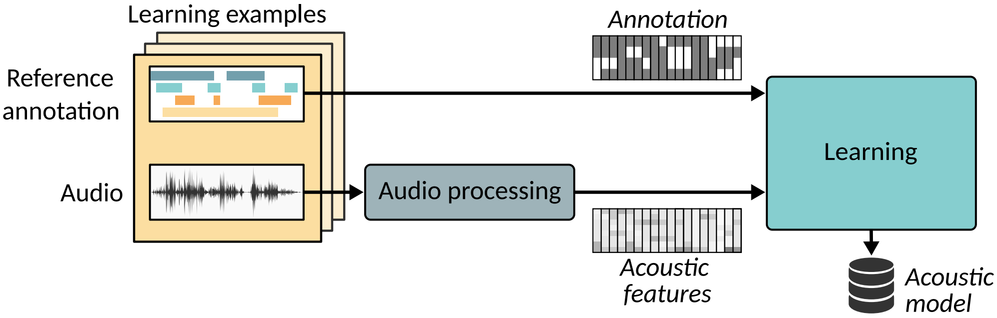


Figure 3.10 Overview of the supervised learning process for audio content analysis. The system implements a multi-class multi-label classification approach for sound event detection task.

is illustrated in Figure 3.10.

The acoustic model partitions the acoustic feature space with a *decision rule* into regions with an assigned class label, and one class label can have multiple disjoint regions associated with it. Recognition is done based on these regions: the observed acoustic features are classified based on which region they fall into. The boundary between regions is called *decision boundary*, and generally most of the misclassifications happen near this boundary. Model learning is guided by the errors or *loss* between the target outputs and estimated outputs from the training examples, and the model parameters are updated through optimization techniques to decrease this error.

The basic assumption behind the learning process is that the test examples will be similar to the training examples, i.e., test example are coming from the same distribution as training examples. The model should be able to robustly estimate the correct output in this situation based on the learned decision rules. In practice, it is very hard to collect a representative set of training examples to cover all potential variability of test data. The *generalization* to unseen examples is the main property of a good classifier, and failing in this is caused by model *overfitting* or *underfitting*. Everyday sounds mostly appear in noisy multi-source situations and have large intra-class variability that leads to high variations in acoustic characteristics (see Section 3.2). Therefore it is challenging to achieve good generalization based on a limited set of training examples. When overfitting, the model has learned peculiarities of the training examples rather than general acoustic characteristics which would generalize well on unseen test examples. This can be caused by a limited set of training examples, or having a too expressive model, e.g. too many parameters given the training material size. When underfitting, the model was not complex enough or not trained sufficiently to be able to robustly classify unseen test examples.

Generally, supervised learning approaches can be categorized into two main types: *generative* and *discriminative*. In generative learning, the underlying class-conditional probability density is estimated explicitly based on learning examples. For each sound class the joint distribution $p(x, y)$ is modeled separately, and the Bayes' rule is used to find the most probable class from which the input was generated by finding the maximum posterior probability $p(y|x)$. Commonly used classifiers for audio content analysis that follow the generative learning approach include Gaussian mixture models (GMM), and hidden Markov models (HMM). In discriminative learning, the modeling concentrates on the boundaries between classes instead of the classes themselves. Data examples are used directly for defining the decision boundaries and finding a direct mapping between inputs to the system and the target outputs [129]. Examples of discriminative learning approaches used for audio content analysis include decision trees, support vector machines (SVM), and neural networks. The work in this thesis concentrates on approaches based on generative learning [P1]–[P7]. Recently, the discriminative learning approaches have gained popularity in audio content analysis primarily because of advancements of deep learning and neural networks, and increased size of available datasets for learning [24, 109].

3.4.2 Recognition

Once the acoustic model is learned, it can be used for classifying sounds into predefined sound classes. An overview of the recognition process is illustrated in Figure 3.11. In the audio processing stage, the acoustic features are extracted for the test audio. The acoustic features are fed as input to the recognition stage, which uses the learned acoustic models to get class-presence information (probabilities or likelihoods depending on the method) for each input feature frame. In the post-processing stage, the class-presence probabilities are converted into class activity, a binary indication of a class being present or not within the current analysis frame. Depending on the target recognition task, the output format of the post-processing will be different. Classification methods can be used for detection by doing classification in short time segments (e.g. one second). The detection task aims to produce classification results in a sufficiently high time resolution which can be processed into onset and offset timestamps describing sound event activity, i.e., sound event.

Classification

In sound classification, class-presence information of multiple short analysis frames from an audio segment to be classified are combined into a single classification output. In a single-label classification task, a single class label is assigned to an item, whereas in multi-label classification, multiple category labels are assigned to a single item.

Frame-level class-presence probabilities can be processed into a single-label classification output either by taking classification decisions at frame-level and combining

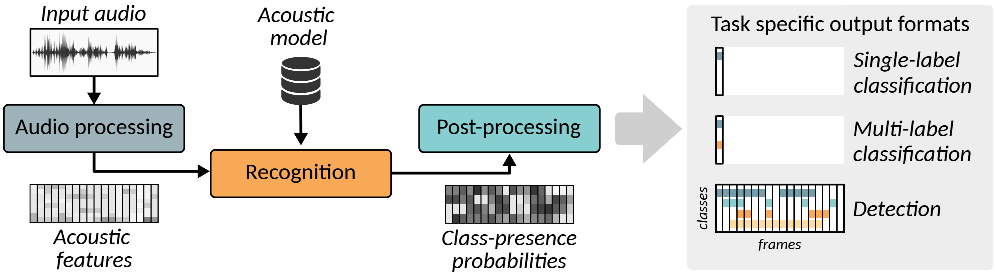

Input audio
Acoustic model
Audio processing
Recognition
Post-processing
Acoustic features
Class-presence probabilities
Task specific output formats
Single-label classification
Multi-label classification
Detection
classes
frames

Figure 3.11 Overview of the recognition process for audio content analysis.

decisions (*hard voting*) or by combining probabilities and making the decision at segment-level (*soft voting*). In the first approach, classification decisions are done at the frame-level by selecting the class with the highest probability, and performing majority voting over these frame-level estimates to get the class with the highest number of occurrences [109]. In the latter approach, the class-presence probabilities are combined either by summing or by averaging, then the class with the highest probability is selected as the classification result [P7]. This approach often results in a higher classification accuracy than hard voting at frame-level, because it gives more weight to more confident estimates.

In multi-label classification, the number of active classes is usually unknown, and one cannot do classification just by selecting the class with the highest probability. The single-label classification scheme can be extended into a multi-label classification either by modifying how the system output is interpreted into activity or by adding extra classes to capture cases when target sound classes are not active. In the first approach, class presence probabilities from frames are summed or averaged similarly to the soft voting scheme, and a probability threshold is used to select the active classes. Probability thresholding to get activity estimates is called *binarization* and it is a common procedure with neural networks [24, 109]. In the second approach, the non-activity of the class is explicitly modeled by introducing extra classes. One extra class, *universal background model* (UBM), can be added to represent the case when no target sound is active [P2]. Another way to model non-activity with extra classes is to create class-wise classifiers such that each of these classifiers is able to recognize only the activity of that particular sound class [P6][P7][22]. Binary classifiers are trained with two classes: a positive class, when the specified sound is active, and a negative class when the specified sound is not active.

Detection

In sound event detection, class presence probabilities of consecutive analysis frames are converted into sound activity information, onset and offset timestamps of the

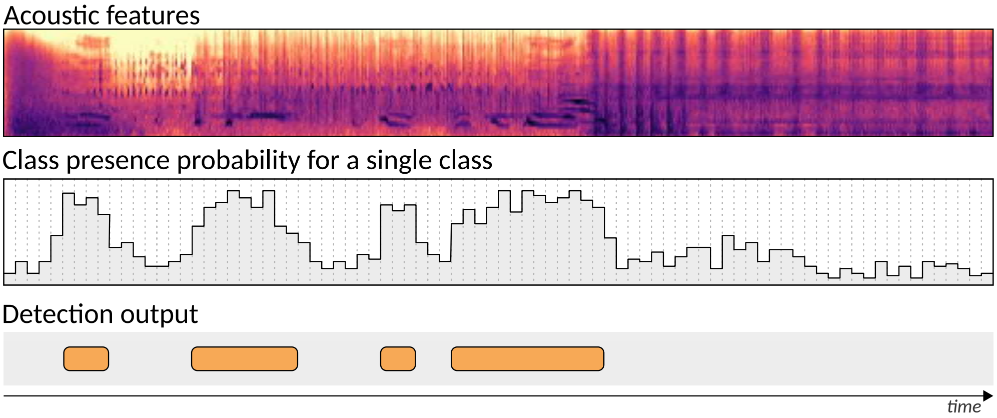


Figure 3.12 Converting class presence probability (middle panel) into sound class activity estimation with onset and offset timestamps (bottom panel).

active sound events. The process is a multi-class multi-label classification task at frame-level, and the aim is to process the classification output into estimates of sound event activity. The process is illustrated in Figure 3.12.

Generally, sound events are much longer than the used analysis window (typical length 20-100 ms) making classification outputs of consecutive analysis frames correlated. Furthermore, everyday sounds have a characteristic temporal structure which is important for the recognition, and this structure spans several analysis frames. The temporal structure is taken into account in the detection stage by using *temporal post-processing* applied to the frame-wise class presence probabilities or the binary class presence estimates. The classification at frame level produces noisy results because feature distributions of sound classes overlap each other and short frames do not have enough information for robust classification. *Median filtering* using a sliding window can be used to filter out this classification noise and to smooth the results from consecutive frames [P7][109]. In this process, the median of binary sound activity estimates within the processing window (e.g. one second worth of analysis frames) is outputted, and the window is shifted one frame forward. The temporal structure of a sound can be addressed in the acoustic model as well by incorporating contextual information in the classification, e.g. using recurrent layers in neural networks [24], which makes post-processing in such approaches less important.

3.4.3 Methods

This section introduces two classification techniques used in this thesis for audio content analysis: Gaussian mixture models, and hidden Markov models. In addition, state-of-the-art classifier techniques based deep learning and specifically deep neural networks are briefly introduced.

Gaussian Mixture Model

The *Gaussian mixture model* is a probabilistic model that can be used to estimate normally distributed sub-populations within the training data in an unsupervised manner. It can be extended into a multi-class supervised classifier by fitting a separate GMM model for training data coming from a specific sound category. In the test stage, class-wise likelihoods (posterior probability $p(y|x)$) are evaluated for an observation, and the class giving maximum likelihood is selected. In sound classification, sub-populations modeled with the mixture model can be seen as variations in acoustic characteristics within a sound category. For example, in the case of a dog barking sound class, the sub-populations can be produced by small, medium, and large dogs with various types of barking as well as different portions of the barking sequence itself.

Before the recent emergence of deep learning, GMM was one of the most commonly used classification methods in sound classification tasks because of its good generalization properties. In automatic speech recognition, it has been used to model individual phonemes, while the transitions from one phoneme to another were modeled with hidden Markov models [145]. In speaker verification, a GMM representing all speakers, a universal background model, was adapted to model the target speaker, with the actual speaker verification done based on the likelihood ratio between the target model and UBM [148]. In music information retrieval, GMMs have been used to classify, for example, musical genre [180] and musical instruments [71].

A GMM estimates the underlying probability density function (pdf) of the observations (acoustic features), and it is capable of representing arbitrarily shaped densities through a weighted mixture of N multivariate Gaussian distributions (also called normal distributions) [42, 137]. The model sums together multiple Gaussian distributions (components), and the whole model is parameterized by the mixture component weights, and the mean and variance of the individual component distributions. The model can be seen as a clustering algorithm with soft assignments, where each modeled data point could have been generated by any of the component distributions used in the model with a corresponding probability, i.e., each mixture component has some *responsibility* for generating a data point. The probability density for an observation $\boldsymbol{x}$ from a class k is computed as

$$p_k(\boldsymbol{x}) = \sum_{n=1}^{N} \omega_n \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_n, \Sigma_n) \tag{3.5}$$

where ω_n is the positive weight of the component n and the normal distribution

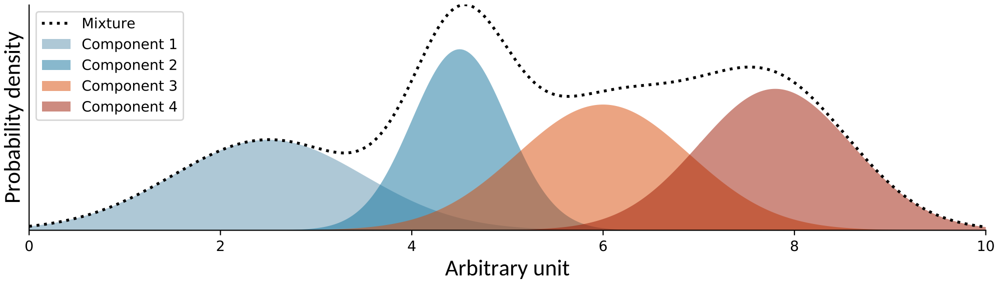


Figure 3.13 Example of one-variate Gaussian distribution with four components.

$\mathcal{N}(x|\mu_n, \Sigma_n)$ is defined by the the mean vector μ_n and the covariance matrix Σ_n. Component weights ω_n sum to unity. Figure 3.13 illustrates the basic principle of the mixture model in the one-variate case. The multivariate Gaussian density function is defined as

$$\mathcal{N}(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \tag{3.6}$$

where d corresponds to the length of the feature vector.

Model parameters ω_n, μ_n, Σ_n in Eq 3.5 are learned based on the training material for each class separately to allow multi-class classification. The training is implemented using the expectation-maximization (EM) algorithm where optimal model parameters are iteratively estimated [35, 104]. A latent variable γ is introduced for each data point to indicate the responsibility of each mixture component for generating a particular data point. The EM algorithm alternates between an expectation step and a maximization step: the first estimating the latent variable, the second updating the model parameters to maximize the likelihood based on the estimated latent variable. The process is often initialized by clustering the data with the k-means algorithm. The number of mixture components N can be optimized as a hyper-parameter using cross-validation, or it can be optimized during the training process as one of the model parameters using information criteria [45, 74]. Depending on the level of intra-class variation and the amount of training data, the best performing N can also vary across classes.

The parameter optimization process is commonly simplified by using diagonal covariance matrices Σ_n to restrict the parameter space. This simplifies the calculations, but at the same time prevents the model from capturing correlations between variables. However, the models trained on decorrelated features perform well in practice. One can use decorrelated acoustic features such as MFCC [P1]–[P7] or use e.g. principal component analysis to decorrelate other acoustic features before using them with a classifier.

In the inference stage, the likelihood of an observation to be generated from each class-wise model is estimated using Eq 3.5. The predicted class is selected based on the maximum likelihood classification principle. As discussed earlier, one can use either hard voting or soft voting to get a single-label classification output for an audio segment containing many analysis frames. With GMMs the usual approach is based on the soft voting scheme, where frame-level likelihoods are accumulated class-wise over the whole segment and the predicted class is selected based on these accumulated likelihoods. The aim is to find class k which has highest likelihood L for the set of observations $X = \boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_M$:

$$L(X;\lambda_k) = \prod_{m=1}^{M} p_k(\boldsymbol{x}_m) \tag{3.7}$$

where λ_k denotes GMM for class k and $p_k(\boldsymbol{x}_m)$ is the probability density function value for observation $\boldsymbol{x}_m$. The general assumption in this approach is that consecutive analysis frames are statistically independent.

Hidden Markov Model

The *hidden Markov model* is a probabilistic temporal model used to model sequential data such as a sequence of acoustic features [145, pp. 321]. HMM relies on a few assumptions: data can be divided into a sequence of stationary time segments called *states*, transitions between the states depend only on the origin and destination (Markov property), and the probability of an observation to be produced by a state does not depend on previous observations.

An HMM is composed of states that respect the previous assumptions. Information on which state produced the output is not observable directly, i.e., the state information is *hidden*. The HMM is defined by the initial state probability distribution Π, state transition probabilities, and output distributions of the states. The probability of being in state i at the beginning of the process is defined by the initial state probability distribution, $\boldsymbol{\Pi} = [\pi_1, \ldots, \pi_i, \ldots, \pi_N]$. Transition probability from state i to state j is defined by a_{ij}, where $i, j = 1, \ldots, N$ and N is the number of states in the HMM. Transition probabilities are presented as $N \times N$ matrix $\boldsymbol{A}$. The observations are the outputs of the states, and the probability of observation x in state j is defined by $b_j(x)$, with parameters of the state distributions collected in $N \times M$ matrix $\boldsymbol{B}$, where M is the number of possible observation symbols. The state output probabilities are based on GMMs, and determined by static and dynamic features [145]. Dynamic features are used to represent the short-time context in HMM, and this can be seen as a heuristic approach to compensate for the assumption of observations being independent [52].

Before the era of deep learning, the standard approach for automatic speech recognition was to model individual acoustic units in speech (phones) with GMMs and the temporal sequence of these units with HMMs [145]. Following the success with speech recognition, HMMs have been successfully applied to many classification and detection tasks in the field of music information retrieval [26] and content analysis of everyday environments. As the time-varying properties of sound are important information for sound identification, the HMMs with temporal modeling abilities have produced good results in many works related to general sound recognition [20, 27, 44, 198] and sound event detection [P1]–[P4].

The topology of the HMM model is defined by the non-zero values in the state transition probability matrix A and it can be represented as a graph having connections a_{ij}. The default topology, a *fully-connected topology*, has all transitions enabled, however, domain knowledge can be used to select the optimal topology for a task by setting some state transitions to zero before model training. The often-used *left-to-right topology* allows only transitions from one state to itself or the next state, the topology being well-suited to model sequential processes that have a clear start, steady part, and end. Such topology is used to model acoustic units in speech, notes played with musical instruments, and everyday sounds. In the case of everyday sounds, hidden states can be seen as modeling different stages of an individual sound event. For example, the sound of a glass cup smashing onto the floor and breaking could be divided into the following three parts suitable for a left-to-right model topology: transient-like sound from the impact, sound produced by a high amount of debris moving on the floor for a short distance, and lastly sound produced by a lower amount of debris moving longer distances on the floor surface. An example of the HMM model represented as a graph can be seen in Figure 3.14.

The HMM model is trained using a special case of the EM algorithm, the Baum-Welch re-estimation algorithm [146]. During the training procedure, the model parameters are iteratively re-estimated while computing the probability that the observed sequence (training data) was produced by the current model. When using HMM for a classification task, each class is represented by a separate HMM, and classification is done with the maximum-likelihood method by finding the model having maximum aposteriori probability for the given observation sequence. Aposteriori probability can be computed using the Viterbi algorithm to find the optimal state sequence, a single best path through the model which provides the highest total likelihood of specific observation sequence being produced by this particular state sequence [192]. In the case of a detection task, like sound event detection, transitions between classes are also important for robust detection. In order to do detection, class-wise HMM models can be merged into a single large HMM and allowing transitions between classes only from certain states. Viterbi algorithm is then used to

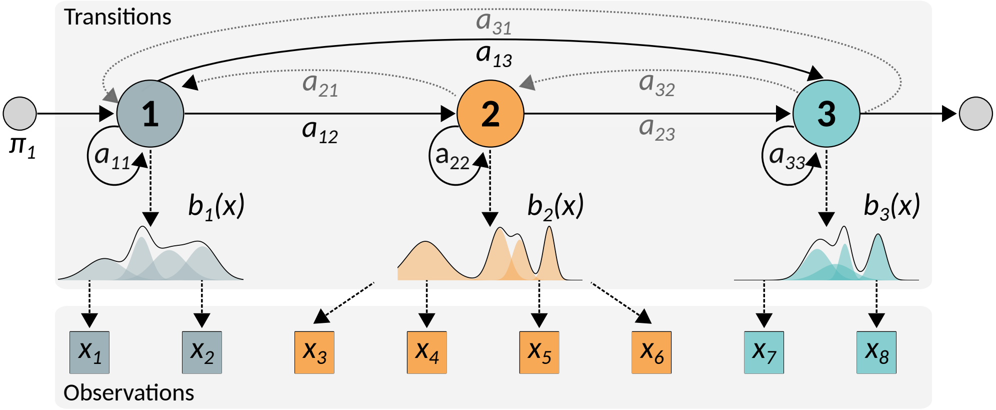


Figure 3.14 Example of a hidden Markov model. The state transition matrix is represented as a graph: nodes represent the states and weighted edges indicate the transition probabilities. Two model topologies are presented in the figure: fully-connected and left-to-right. The dotted transitions have zero-probability in the left-to-right topology.

find the optimal path through this large HMM, and the detection output is produced based on the found optimal state sequence and the sequence of classes these states belong to.

Deep learning

The state-of-the-art computational audio content analysis systems are nowadays almost without exception based on deep learning approaches, and they are showing exceptional performance over classical machine learning approaches. The work included in this thesis was carried out before the era of deep learning and does not utilize any deep learning. However, a brief introduction to deep learning is given here to provide the reader with a modern perspective.

The data processing in deep learning is inspired by information processing in the human brain during the process of acquiring new knowledge and the process of recalling stored knowledge. The main idea of the data processing in deep learning is to model complex concepts, such as sound event classes, by combining simpler and more abstract concepts learned automatically from data. The modeling is done using an artificial neural network, a deep multilayered structure, where each layer is constructed from multiple artificial neurons, and neurons between layers are connected. Deep learning is used for classification as a discriminative learning approach, and robust modelling is ensured by using a large amount of learning examples.

An artificial neuron is an elementary unit of artificial neural networks. A neuron has n inputs and each input has a weight parameter w_i. The weighted sum of the inputs summed with a constant bias factor b is passed through an activation function

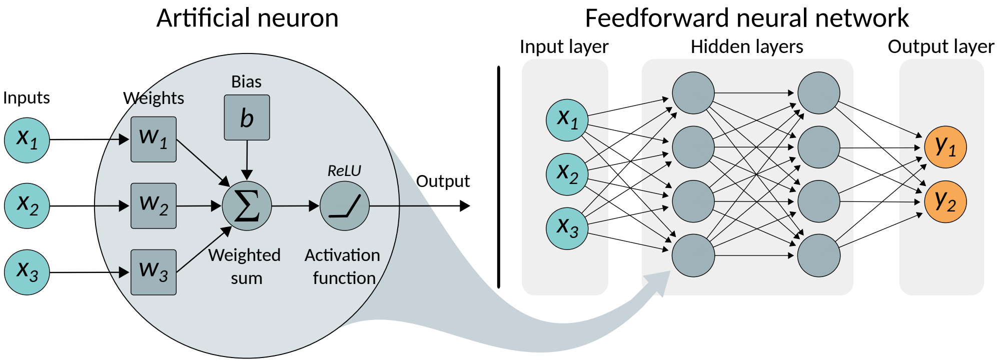


Figure 3.15 Overview of an artificial neuron (left panel) and the basic structure of a feedforward neural network (right panel).

to produce the neuron's output. By using a non-linear activation function, the neural network can model highly complex relationships in the data. A classical example of an artificial neural network is the *feedforward neural network* (FNN) where information between layers flows only from the input towards the output without feedback connections [62]. The basic inner-structure of an artificial neuron and the feedforward network structure is shown in Figure 3.15.

A network consists of an *input layer*, an *output layer*, and a number of intermediate layers called *hidden layers* for which the training data does not provide desired outputs. During the learning process, parameters w_i and b for each neuron in the network are iteratively optimized to produce the desired target outputs (reference labels) at the output layer given the input data (acoustic features) to the network. The learning process uses the back-propagation algorithm; at each learning iteration, a loss function is calculated between the output of the network and the target output for the learning examples and this loss is fed back through the network and the network parameters are adjusted to lower the loss for the next iteration. The non-linear optimization method called gradient descent is used to update the network parameters in the opposite direction of the gradient of the loss function after a small set of learning examples (a batch). The activation function for neurons in the output layer is selected based on the classification task; for a single-label classification, softmax activation is commonly used, whereas for a multi-label classification sigmoid activation is used. The hidden layers often use rectified linear units (ReLU) [60] as activation function.

The feedforward neural network was the first deep learning approach successfully used for audio content analysis applications, and it showed clear improvement in performance over the established GMM and HMM-based approaches, for example, in acoustic scene classification [109] and sound event detection [23, 59, 109]. Short temporal context can be taken into account in FNN-based systems by using context windowing at network input (see Section 3.3.2). Two main problems with FNNs for audio content analysis are their sensitivity to variations in time and frequency due to fixed connections between the input and the hidden layers of the network, and their inability to model long temporal structures essential for the recognition of some sound events.

Convolutional neural networks (CNN) are nowadays chosen over FNNs as they generally produce more robust models [123, 124]. CNN is a time and frequency shift-invariant model designed to take advantage of the 2D structure of the input [94]. This is achieved by utilizing the local connectivity inside the network to allow parts of the network to specialize in different high-level features of the data during the learning process. The final classification output is produced based on the learned high-level features using a few feedforward layers before the output layer. Convolutional recurrent neural networks (CRNN) are often used for sound event detection [2, 24, 203]. These networks resemble CNNs, but they add layers containing feedback connections that capture long temporal structures in the data.

4 SOUND EVENT DETECTION IN EVERYDAY ENVIRONMENTS

Detection of sound events is required to gain an understanding of the content of audio recordings from everyday environments. Sound events encountered in our everyday environments are often overlapping other sounds in time and frequency, as discussed in the previous chapters. Therefore, polyphonic sound event detection is essential for well-performing audio content analysis in everyday environments. This chapter goes through the work published in [P1]–[P7]. These publications are dealing with various aspects of polyphonic detection system: forming audio datasets, evaluating the detection performance, training acoustic models from mixture signals, detecting overlapping sounds, using contextual information, and organizing evaluation campaigns related to sound event detection.

Problem Definition

Sound event detection aims to simultaneously estimate *what* is happening and *when* it is happening. In other words, the aim is to automatically find a start and end time for a sound event and associate a textual class label for the sound event. The detection can be done either by outputting the most prominent sound event at the time (monophonic detection) or by outputting also other simultaneously active events (polyphonic detection). Examples of both types of detection are shown in Figure 4.1. Monophonic detection captures a fragmented view of the auditory scene, long sound events could be split in the detection into smaller events, and quieter event in the background might get covered by louder events and not get detected at all. Sound events detected with monophonic detection scheme might be sufficient for certain applications, however, for general content analysis, the polyphonic detection is often required.

Input to the detection system is acoustic features $\boldsymbol{x}_t$ which are extracted in each time frame t for the input signal. Aim is to learn an acoustic model able to estimate presence of predefined sound event classes at each time frame $\boldsymbol{y}_t$. The model learning is done based on learning examples: audio recordings along with annotated sound event activities. The sound event class presence probability at the time frame is

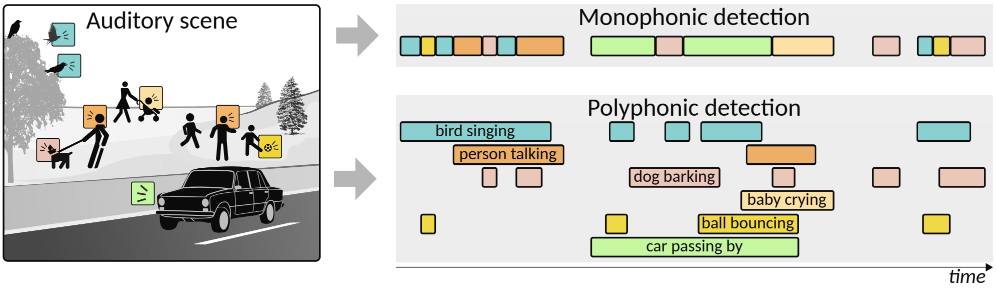


Figure 4.1 Illustration on how monophonic and polyphonic sound event detection captures the events in the auditory scene.

given by posterior probability $p(y_t|x_t)$. The probabilities are converted into binary class activity per frame, *event roll*, and sound event onset and offset time stamps are acquired based on consecutive active frames. The output of the system is usually formatted as a list of detected events, *event list*, containing a class label with onset and offset timestamps for each event.

Challenges

The main challenges faced in the development of a robust polyphonic SED are related to characteristics of everyday environments and everyday sounds. When the main part of the work included in this theses was done, the additional challenges were related to the dataset quality, amount of examples in the dataset, and lack of established benchmark datasets and evaluation protocols to support the SED system development.

Simultaneously occurring sound events in the auditory scene produce a mixture signal, and the challenge of the SED system is to be able to detect individual sound events from this signal. The detection should be focusing only on certain sound event classes while being robust against the *interfering sounds*. As the number of sound event classes that can be realistically used in the SED system is much lower than the actual number of sound sources in most of the natural auditory scenes, some amount of these overlapping sounds will be always unknown to the SED system and can be considered to be interfering sound. In a real use case, the position of the sound-producing source in relation to the audio capturing microphone cannot be controlled, leading to varying loudness levels of the sound events and together with overlapping sounds challenging signal-to-noise ratios in the captured audio recordings. The variability in the acoustic properties of the environments (e.g. room acoustics, or reverberation), will further contribute to the diversity of audio material.

Sound instances assigned with the same sound event label have often a large intra-class variability. This is due to the variability in the sound-producing mechanisms of the sound events, and this variability should be taken into account when learning the acoustic models for the sound events to produce a well-performing SED system. Ideally, one needs a large set of learning examples to fully cover the variability of the sounds in the model learning, however, in practice for most use cases it is impossible to collect such dataset. Thus the robust acoustic modeling of the sound events with limited amount of learning examples is one of the major challenges. Sound events occurring in natural everyday environments are connected through the context they appear, however, the temporal sequence of these events seldom follows any strict structure. This unstructured nature of the audio presents an extra challenge for the SED system design compared to speech recognition or music context retrieval systems where the analysis can be steered with structural constrains of the target signal.

The development of the robust SED system is dependent on the dataset quality and the number of examples per sound event class available in the dataset. The collection of data is relatively easy, however, manual annotation is a subjective process and this poses a challenge for the learning and evaluation of the system. In the annotation stage, the person is listening to the recordings and manually indicating the onset and offset timestamps of the sound events, and selecting the appropriate textual label to describe the sound event. As the amplitude envelopes of the sound events are often quite smooth, clear change points are hard to determine and this temporal ambiguity will eventually lead some degree of subjectivity in the onset/offset annotations. Furthermore, the selection of the textual label involves the listener's own prior experiences, and if free label selection is allowed, sounds will be labeled in a varying manner across annotators. Even the same annotator might label similar sounds differently depending on the context the sound event occurred. These subjective aspects of the annotation process produce noisy reference data, which has to be taken into account in the development and evaluation of the SED systems.

4.1 Related Work

The research effort related to audio content analysis in everyday environments has been steadily increasing over the last ten years. The works in this field can be divided into five major research topics: acoustic scene classification (ASC), sound event classification, audio tagging, sound event detection, and joint sound event localization and detection (SELD). The research history of this field together with the works included in this thesis are illustrated on the timeline in Figure 4.2. This section discusses the related work including research tasks, approaches, evaluation campaigns, and milestones.

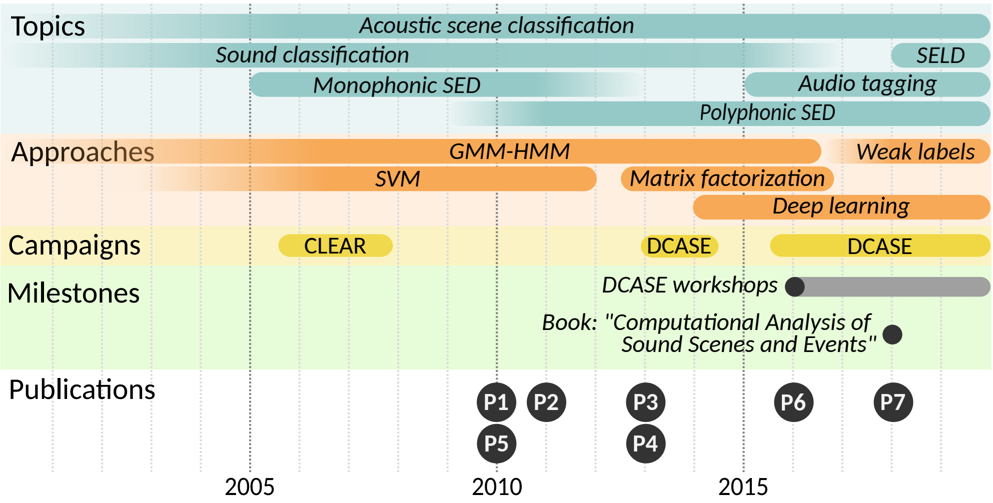


Figure 4.2 Research timeline in the field of audio content analysis in everyday environments indicating major trends in research topics and approaches and highlighting evaluation campaigns and major milestones.

Research Field

Audio content analysis in everyday environments has existed as a separate and identifiable research field only in the last 10-15 years. Preliminary research work falls mostly under Computational Auditory Scene Analysis (CASA), where computational methods mimicking the human auditory system are used to derive properties of individual sounds in mixture signals and group them into sound events [43, 168]. The early research in the field is inspired predominantly by speech recognition and music information retrieval, and the approaches are based on the traditional supervised machine learning methods such as GMM [90], HMM [211], and SVM [212]. The contemporary work is based on deep learning [24], and the approaches are influenced strongly by general machine learning development across major research domains such as speech recognition, computer vision, and natural language processing. Whereas the early works utilized limited sized strongly labeled datasets in the training and evaluation, the contemporary works employ substantially more extensive datasets containing possibly weakly labeled data.

Evaluation campaigns have had a substantial role in the growth of the research field in recent years by standardizing the evaluation protocols, establishing evaluation metrics, publishing open benchmark datasets, and providing an easy access platform for researchers from neighboring research fields. The first evaluation campaign to include tasks related to the analysis of everyday environments was the Classification of Events, Activities, and Relationships (CLEAR) Evaluation organized in 2006 and 2007 [163]. The campaign presented monophonic sound event detection and sound event classification tasks in the meeting room environment while using a multi-microphone setup [162, 171]. A community-driven international challenge, the Detection and Classification of Acoustic Scenes and Events (DCASE), was organized first in 2013 [166] and has been organized annually since 2016 [P7]. The challenge attracts annually 200-400 system submissions from 60-130 international research teams. The new topics are being introduced at each edition to spark new research and to foster ongoing research in the research field. Acoustic scene classification and sound event detection have been the core topics for each challenge edition, and different research questions related to these topics have been addressed by varying the task setups. The DCASE challenge uses public datasets, and these datasets have gained extensive popularity outside the evaluation campaigns as well.

The annual DCASE Workshop, organized since 2016, has had an instrumental role in the growth of the research field by providing a focused and peer-reviewed publication platform for the community [101, 142, 189, 190]. The first and currently the only book to cover topics broadly in the research field was "Computational Analysis of Sound Scenes and Events" edited by T. Virtanen *et al.* [191]. The book can be considered as one of the milestones in the field, as it exclusively focuses on content analysis of environmental audio and comprehensively goes through research topics related to it while describing the current state-of-the-art methods.

Everyday Sound Recognition

Works related to the recognition of everyday sounds can be categorized roughly into two groups; ones aiming at recognizing acoustic scenes, and ones aiming at recognizing individual sounds or sound events.

Acoustic scene classification is a task where a textual label identifying the environment is assigned to an audio segment [9]. In addition to ASC, the task is referred in the literature as computational auditory scene recognition [138] or as audio-based context recognition [44]. The main application for the ASC is context awareness, the ability to determine the context around the device, and to self-adjust the operation mode of the device accordingly. The task can be set either as a closed-set classification task where all scene classes are assumed to be known in advance or an open-set classification task where unknown scene classes may be encountered as well while the system is running [124]. Early approaches for the task were based on traditional supervised machine learning methods such as GMM [138] and HMM [44] using spectral features such as MFCCs. DCASE2013 introduced ASC as an evaluation campaign task [166]. A clear trend emerging among the best performing submissions for the task was the usage of the temporal information in the acoustic features within the medium-long time segments (400 ms - 4 s) as features [57, 147, 150]. Deep learning approaches surfaced by the DCASE2016 challenge, where almost half of the submission used FNNs, CNNs, or RNNs [105]. However, due to the limited datasets at the time, the classical learning methods such as SVM and factor analysis methods such as NMF performed respectably well against deep learning methods. Along with the bigger datasets introduced in the later DCASE challenges, deep learning methods have outperformed the classic approaches with a clear margin. State-of-the-art datasets contain recordings captured with a few types of recording devices simultaneously in a large selection of locations. Current state-of-the-art systems usually utilize the convolutional neural networks as an acoustic model to take advantage of the 2D structure of the time-frequency representation of the audio input [1]. Furthermore, these systems tend to use a multitude of *data augmentation* techniques to diversify the learning examples and to produce a more robust acoustic model capable of dealing with different recording conditions, devices, and locations. The current state-of-the-art systems can surpass human recognition accuracy in individual cases where the test subject is not familiar with the specific environment (so-called non-expert listener) [117].

Sound classification and tagging task aims to assign one or multiple textual labels to an audio segment, labels that describe the sound-producing events active within the segment. In case only a single label at a time can be assigned to an audio segment, the task is usually referred to as a *sound event classification* or *environmental sound classification*. The task is closely related to sound event detection, as the tagging or classification in short time-segments can be easily used to produce detection output. For example, sound event tagging and classification can be applied in audio based monitoring systems within fixed time-segments to produce activity information of specific sound events [12]. Early works on this topic focused on a small selection of sound classes related to a single environment such as kitchen [88], bathroom [29], office [178], or meeting room [172], whereas the more recent works have expanded both the selection of classes and environments used in the development [76, 141]. Techniques for the task follow the same main trends as for acoustic scene classification. The early works are based on classic machine learning approaches such as SVM and HMM, while the more recent works are based on deep learning methods such as CNN and CRNN. Some of the recent works have moved away from traditional handcrafted features such as MFCCs or mel-band energies by applying feature learning techniques [153] or using an end-to-end classification scheme [176]. Similarly to ASC, data augmentation techniques are widely used in the state-of-the-art systems [154].

Sound Event Detection

Sound event detection is a task where sound events are temporally located and classified in an audio signal at the same time: for each detected sound event instance an onset and offset are determined and a textual label describing the sound is assigned. Works related to sound event detection can be roughly categorized based on the complexity of the system output into monophonic and polyphonic sound event detection approaches. The main applications for sound event detection include audio-based monitoring, multimedia access, and human activity analysis. These applications are very much related to general audio content analysis applications discussed earlier in Section 1.1. Event detection can be applied in monitoring and surveillance systems that require detailed information about event onset and offset timestamps in addition to event labels [33]. Example use cases for monitoring are the detection of specific events such as glass breaking [109], gunshots [30], screams [185], or footsteps [6]. As content-based analysis for multimedia, sound event detection can be used to automatically generate keywords [80, 169] for large repositories and further enable content-based indexing and search functionality. In human activity analysis, SED can be used to detect individual sound events which are associated with certain activity [28]. Research done in sound event detection also supports and complements the research in many neighboring fields such as bioacoustics where it is used to assess the biodiversity [53], and in robotics to facilitate human-robot interaction [38].

Early approaches for sound event detection were based on traditional pattern recognition techniques like GMM [183, 200], HMM [73, 211], and SVM [170, 173]. These works usually focused on cases where sound events were encountered mostly as a sequence of individual sounds with silence or background noise between them, and use cases with a relatively small number of classes. Many of these works are related to early evaluation campaigns such as CLEAR 2006 and 2007 [164, 171], and DCASE2013 [166]. Starting about one decade ago, the specific tasks tackled in SED became more diverse and complex: the number of event classes has increased [P1], detection of overlapping sound event has been studied using synthetic [93] and real data [P7], and imbalanced data has been studied as a task of rare sound event detection [109].

The more recent works are based on deep learning methods such as CNN [87] and CRNN [24], and similarly to recent sound recognition approaches, data augmentation techniques [136] are often used to increase data diversity. Commonly these works use handcrafted acoustic features such as mel-band energies, while some work has been done to learn automatically better representations from the spectrogram [153] or directly from the raw audio signal [21]. As an alternative to data augmentation, transfer learning techniques are used to cope with the lack of sufficient data. In these techniques, a neural network based model is trained to solve a pretext task using large amount of data, then the outputs of the pretrained model are used to produce new features, *embeddings*, to be used in the actual target task [32, 82]. The deep learning-based approaches have highlighted a need for large open-access datasets. First sound event datasets were individual research group efforts ([P6] and [106])

Table 4.1 Information about the datasets used in this thesis.

		Meta data			Audio data		
Dataset	Used in	Event instances	Event classes	Scenes classes	Files	Length	Notes
Sound Effects 2009	[P1]	1359	61	9	1359	9h 24min	Isolated sounds Proprietary dataset
TUT-SED 2009	[P1]–[P5]	10040	61	10	103	18h 53min	Continuous recordings Proprietary dataset
TUT-SED 2016 Development	[P6] and [P7]	954	18	2	22	1h 18min	Continuous recordings Open dataset
TUT-SED 2016 Evaluation	[P7]	511	18	2	10	35min	Continuous recordings Open dataset

containing manually annotated event onsets and offsets. As the datasets got larger, such detailed annotations became impractical to produce. Most recent datasets contain only recording-level annotations of sound event activity (weak annotation), which allows producing annotated data easier at larger volumes. Instead of using expert annotators, the annotation effort has been further decreased by using crowd-sourced non-expert annotators [47], possibly automatically tagging segments and get them verified by annotators [58]. Data collected this way can be considered noisy because some event labels may be incorrect or missing, and weak because it does not contain onset and offset timestamps for the sound events. Learning from such data requires special techniques to compensate for the unreliability of the event labels [47], and weakly-supervised learning approaches [91, 203] for being able to train detection system.

4.2 Audio Datasets

The work included in this thesis is based on three datasets: Sound Effects 2009, TUT-SED 2009, and TUT-SED 2016. All of these datasets were collected and annotated for sound event detection research in the Audio Research Group at Tampere University. The information about these datasets is summarised in Table 4.1.

Sound Effects 2009

For work in [P1], a collection of isolated sounds was collected from a commercial online sound effects sample database (Sound Ideas samples through StockMusic.com). The samples were originally captured for commercial audio-visual productions in a close-microphone setup having relatively minimal background ambiance presence. Samples were collected from nine general contextual classes: crowd, hallway, household, human, nature, office, outdoors, shop, vehicles. In total the dataset comprises 1359 samples belonging to 61 distinct classes.

TUT-SED 2009

The TUT-SED 2009 dataset was the first sound event dataset having real-life continuous recordings captured in large amount of common everyday environments. The dataset was manually annotated with strong labels. This dataset was used in the majority of the works included in this thesis [P1]–[P5]. The dataset is a proprietary data collection, and cannot be shared outside the University of Tampere. The data was originally collected as part of an industrial project where the public release was never the aim. As a result, permissions for public data release were not asked from persons present in the recorded scenes. The aim of the data collection was to have a representative collection of audio scenes, and recordings were collected from ten acoustic scenes. Typical office work environments were represented in the data collection with *office* and *hallway* scenes. The *street*, *inside a moving car*, and *inside a moving bus* scenes represented typical urban transportation scenarios, whereas the *grocery shop* and *restaurant* scenes represented typical public space scenarios. Examples of leisure time scenarios were represented by *beach*, *in the audience of basketball game*, and *in the audience of track and field event* scenes.

For each scene type, a single location with multiple recording positions (8-14 positions) was selected as the aim of the data collection was to see how well material from a tightly focused set of acoustic scenes could be modeled. Each recording was 10-30 minutes long to capture a representative set of events in the scene. In total, the dataset consists of 103 recordings totaling almost 19 hours. The audio recordings were captured using a *binaural* recording setup, where a person is wearing in-ear microphones (Soundman OKM II Classic/Studio A3) in his/her ears during the recordings. Recordings were stored in a portable digital recorder (Roland Edirol R-09) using a 44.1kHz sampling rate and 24-bit resolution.

All the recordings were manually annotated by indicating onset and offset timestamps of events and assigning a descriptive textual label for the sound events. The annotations were done mostly by the same person who did the recordings to ensure as detailed as possible annotations: the annotator had some prior knowledge of the auditory scene to help identify the sound sources. A low-quality video was captured during the audio capture to help annotation of complex scenes with a large variety of sound sources (e.g. street environment) by helping the annotator to recall the scene better while doing the annotation. Due to the complexity of the material and the annotation task, the annotator first made a list of active events in the recording, and then annotated temporal activity for these events within the recording. The event labels for the list were freely chosen instead of using predefined set of global

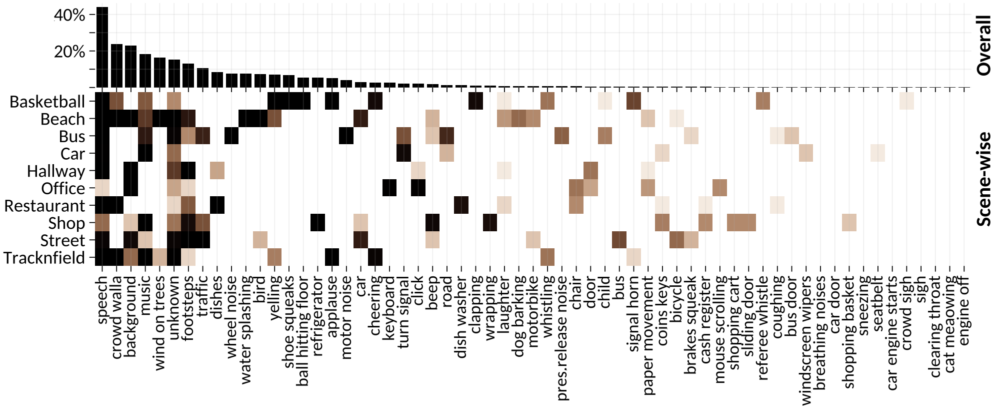


Figure 4.3 Event activity statistics for TUT-SED 2009 dataset. Event activity is presented as percentage of active event time versus overall duration. Upper panel shows overall event activity, while lower panel shows scene-wise event activity.

labels. This resulted in a large set of labels which were then manually grouped into 61 distinct event classes after the whole dataset was annotated. On average, there was 2.7 simultaneous sound events active at all times in the recordings. In the grouping process, labels describing the same or very similar sound events were pooled under the same event class, for example, "cheer" and "cheering", and "barcode reader beep" and "card reader beep". Only event classes containing at least 10 examples were taken into account, while more rare events were collected into a single class labeled as "unknown". Figure 4.3 illustrates the relative amount of event activity per class for the whole dataset as well as per scene. Each scene class has 14 to 23 active events, and many event classes appear in multiple scenes (e.g., speech), while some event classes are highly scene-specific (e.g., referee whistle in basketball games). For example, "speech" events covers 43.9% of the recorded time in the dataset. Overall, the activity amount of event classes is not well-balanced as expected for natural everyday environments.

TUT-SED 2016

The creation of the TUT Sound events 2016 dataset (TUT-SED 2016) was motivated by the lack of an *open dataset* with high acoustic variability [P6]. The data collection was implemented in 2015-2016 under the European Research Council funded Everysound project, and the recording locations were selected from Finland. To ensure high acoustic variability of the captured audio each recording was done in a different location: different streets, different homes. Compared to the TUT-SED 2009 dataset, the TUT-SED 2016 dataset has two scene classes (indoor home environments and outdoor residential areas) and larger acoustic variability within the scene classes. The recording setup was similar to TUT-SED 2009, binaural in-ear microphone setup with the same microphone model and digital recorder device with the same format settings (44.1 kHz and 24-bit). The duration of recordings were set to 3-5 minutes considering this would be the most likely length that someone would record in a real use case. The person recording was required to keep the body and head movement minimum during the recording to enable the possible use of spatial information present in binaural recordings. Furthermore, the person was instructed to keep the amount of his/her own speech to a minimum to avoid near-field speech.

The sound events in the recordings were manually annotated with the onset and offset timestamps, and freely chosen event label. A noun-verb pair was used as an event label during the annotation (e.g. "people; talking" and "car; passing"); nouns were used to characterize the sound source while verbs to characterize the sound production mechanism. Recordings and annotations were done by two research assistants, each annotated the material he/she recorded and they were instructed to annotate all audible sound events in the scene. In the post-processing stage, recordings were annotated for microphone failures and interference noises caused by mobile phones, and this was stored as extra meta information. Sound event classes used in the published dataset were selected based on their frequency in the raw annotations and the number of different recordings they appeared in. The event labels that were semantically similar given the context were mapped together to ensure distinct classes. For example, "car engine; running" and "engine; running" were mapped together, and various impact sounds such as "banging" and "clacking" were grouped together under "object impact". This resulted in a total of 18 sound classes, each having sufficient amount of examples for learning acoustic models.

The dataset was used in DCASE Challenge 2016 task for sound event detection in real life audio [P7], and the data was released in two datasets: *development dataset* and *evaluation dataset*. The development dataset was bundled with 4-fold cross-validation setup, while the evaluation dataset was originally released without reference annotations and later updated with reference annotations after the evaluation campaign.

During the data collection campaign a large amount of recordings from 10 scene types were collected, and only a small subset was published in the TUT-SED 2016 dataset [116, 120]. Later, more material was annotated in a similar fashion, and released as TUT-SED 2017 dataset [121, 122] for the DCASE challenge 2017 [109]. This dataset contained recordings from a single scene class (street), and had relatively small number of target sound event classes (6). The rest of the material was released without sound event annotations as datasets for acoustic scene classification tasks: TUT Acoustic Scenes 2016 (TUT-ASC 2016) [114, 115] for DCASE challenge 2016 [P6] and TUT Acoustic Scenes 2017 (TUT-ASC 2017) [118, 119] for DCASE challenge 2017 [109].

4.3 Evaluation

The quantitative evaluation of the SED system performance is done by comparing the system output with a reference available for the test data. For the datasets used in this thesis, the reference was created by manually annotating the audio material (see Section 3.2.2) and storing the annotations as a list of sound event instances with associated textual label and temporal information (onset and offset timestamps). The evaluation takes into account both the label and the temporal information. When dealing with monophonic annotations and monophonic SED systems, the evaluation is straightforward as the system output at a given time is correct if the predicted event class coincides with the reference class. However, in the case of polyphonic annotations and polyphonic SED systems, the reference can contain multiple active sound events at a given time and there can be multiple correctly and erroneously detected events at the same time instance. All these cases have to be counted for the metric. The evaluation metrics for polyphonic SED can be categorized into *segment-based* and *event-based* metrics depending on how the temporal information is handled in the evaluation. This section is based on work published in [P2], [P3] and [113], and it addresses the research question Q3 (see Section 1.2) on how to evaluate sound event detection systems with polyphonic system output.

4.3.1 Segment-Based Metrics

The first segment-based metric for polyphonic sound event detection, called block-wise F-score, was introduced and used in [P2] and [P3]. In [113] this metric was formalized as *segment-based F-score*, and *segment-based error rate (ER)* was introduced to complement it. These metrics have since become the standard metrics in the research field, and they have been used in many DCASE challenge tasks as ranking criteria. In this thesis, the segment-based F-score is used as performance measurement in [P2]–[P4], [P6] and [P7], and segment-based ER is used as metric in [P6] and [P7]. In the segment-based evaluation, the intermediate statistics for the metric are calculated in a fixed time grid, often in one-second time segments. An illustrative example showing metrics calculation is shown in Figure 4.4.

The sound event activity is compared between the reference annotation and the system output in fixed one-second segments. The event is considered correctly detected if both reference and output indicate event activity and this case is referred to as *true positive*. In case the system output indicates an event to be active within the segment but the reference annotation indicates the event to be inactive, the output is considered as a *false positive* within the time segment. Conversely, in case the reference indicates the event to be active within the segment, and the system output indicates

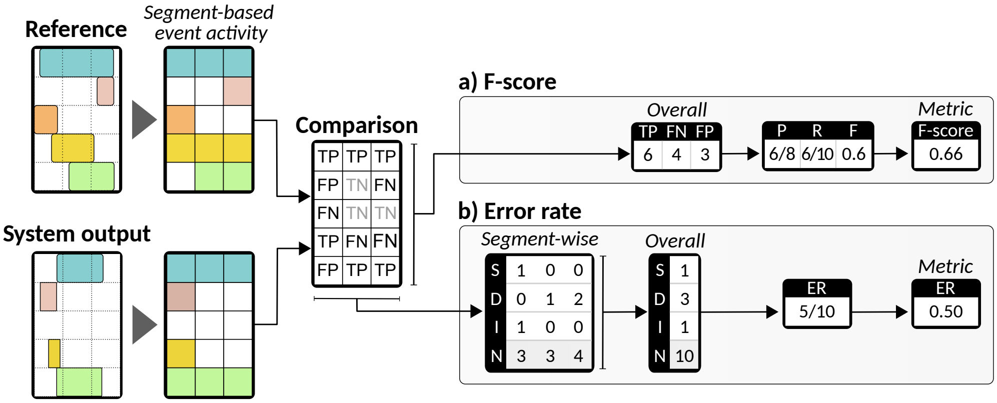


Figure 4.4 Calculation of two segment-based metrics: F-score and error rate. Comparisons are made at a fixed time-segment level, where both the reference annotations and system output are rounded into the same time resolution. Binary event activities in each segment are then compared and intermediate statistics are calculated.

inactivity for the same event class, the output is considered as a *false negative*. Total counts of true positives, false positives and false negatives are denoted as TP, FP, and FN.

F-score

Segment-based F-score is calculated by first accumulating the intermediate statistics over evaluated segments for all classes and then summing them up to get overall intermediate statistics (instance-based metric, micro-averaging). The precision P and recall R are calculated according to the overall statistics as

$$P = \frac{TP}{TP+FP}, \quad R = \frac{TP}{TP+FN} \tag{4.1}$$

and the F-score:

$$F = \frac{2 \cdot P \cdot R}{P+R} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{4.2}$$

The calculation process is illustrated in panel (a) of Figure 4.4.

The F-score is a widely known metric and easy to understand, and because of this, it is often the preferred metric for SED evaluation. The magnitude of the F-score is largely determined by the number of true positives, which is dominated by the system performance on the large classes. In this case, it may be preferable to use class-based averaging (macro-averaging) as overall performance measurement, which means calculating F-score for each class based on the class-wise intermediate statistics, and then averaging the class-wise F-scores in order to get a single value. However, this requires the presence of all classes in the test material to avoid classes with undefined recall ($TP + FN = 0$). This calls for extra attention when designing experiments in a train/test setting, especially when using recordings from uncontrolled everyday environments. In this thesis the segment based F-score is used with two different segment lengths, and is denoted as $F_{seg,1sec}$ for 1-second segments lengths and as $F_{seg,1sec}$ for 30-second lengths.

Error Rate

Error rate (ER) measures the amount of errors in terms of *substitutions* (S), *insertions* (I), and *deletions* (D) that are calculated in a segment-by-segment manner. In the metric calculation, true positives, false positives, and false negatives are counted in each segment and based on these counts substitutions, insertions, and deletions are then calculated in each segment. In a segment k, the number of substitution errors $S(k)$ is defined as the number of reference events for which the system outputted an event but with an incorrect event label. In this case, there is one false positive and one false negative in the segment; substitution errors are calculated by pairing false positives and false negatives without designating which erroneous event substitutes which event. Once the substitution errors are counted per segment, the remaining false positives are counted as insertion errors $I(k)$ and false negatives as deletion errors $D(k)$. The insertion errors are attributed to segments having incorrect event activity in the system output, and the deletion errors are attributed to segments having event activity in the reference but not in the system output. This can be formulated as follows:

$$\begin{aligned} S(k) &= \min(FN(k), FP(k)) \\ D(k) &= \max(0, FN(k) - FP(k)) \\ I(k) &= max(0, FP(k) - FN(k)) \end{aligned}$$

The error rate is calculated then by summing the segment-wise counts for S, D, and I over the total number of evaluated segments K, with $N(k)$ being the number of active reference events in segment k [143]:

$$ER = \frac{\sum_{k=1}^{K} S(k) + \sum_{k=1}^{K} D(k) + \sum_{k=1}^{K} I(k)}{\sum_{k=1}^{K} N(k)} \tag{4.3}$$

The metric calculation is illustrated in panel (b) of Figure 4.4.

The total error rate is commonly used to evaluate the system performance in speech recognition and speaker diarization evaluation, and parallel use in SED makes the metric more approachable for many researchers. On the other hand, interpretation of the error rate can be difficult as the value is a score rather than a percentage and the value can be over 1 if the system makes more errors than correct estimations. An error rate with exact value of 1.0 is also trivial to achieve with the system outputting no active events. Therefore, additional metrics, such as segment-based F-score, should be used together with ER to get a more comprehensive performance estimate for the system.

4.3.2 Event-Based Metrics

The event-based F-score and error rate are used as metrics in [P6]. In these metrics, the system output and the reference annotation are compared in an event-by-event manner: intermediate statistics (true positives, false positives, and false negatives) are counted based on event instances. In the evaluation process, an event in the system output is regarded as correctly detected (true positive) if it has a temporal position that is overlapping with the temporal position of an event in the reference annotation with the same label, and its onset and offset meet specified conditions. An event in the system output without correspondence to the reference annotation according to the onset and offset condition is regarded as a false positive, whereas the event in the reference annotation without correspondence to system output is regarded as a false negative.

For the true positive, the positions of event onsets and offsets are compared using a temporal *collar* to allow some tolerance and set the desired evaluation resolution. The manually created reference annotations have some level of subjectivity in the temporal positions of onset and offset (see Section 3.2.2) and the temporal tolerance can be used to alleviate the effect of this subjectivity in the evaluation. In [P6], a collar of 200 ms was used, while a more permissive collar of 500 ms was used, for example, in DCASE challenge task for rare sound event detection [109]. The offset condition is set to be more permissive as the exact offset timestamp is often less important than the onset of well-performing SED. The collar size for the offset condition adapts to different event sizes by selecting maximum among the fixed 200 ms collar and the 50% of the current reference event duration to cover the differences between short and long events. Evaluation of event instances based on these conditions is shown in Figure 4.5. The evaluation can be done solely based on onset condition or based on both onset and offset conditions depending on how the system performance is required to be evaluated. In [P6], both conditions are used together.

The event-based F-score is calculated the same way as the segment-based F-score. The event-based intermediate statistics (TP, FP, and FN) are counted and summed up to get overall counts. Precision, recall, and F-score are calculated based on Equations 4.1 and 4.2. Same as for segment-based F-score, event-based F-score can be

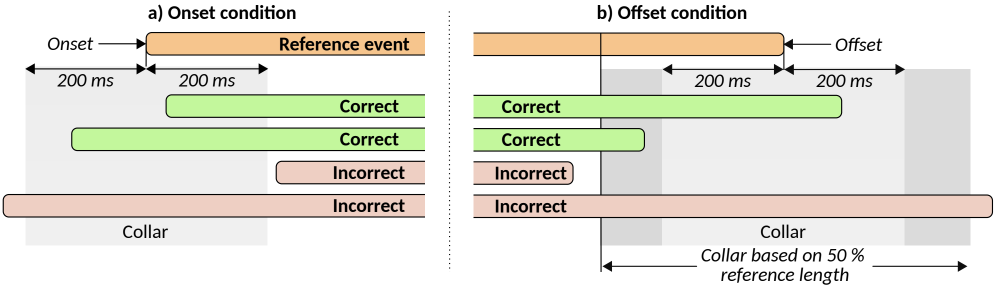


Figure 4.5 The reference event and events in the system output with the same event labels compared based on onset condition and offset condition.

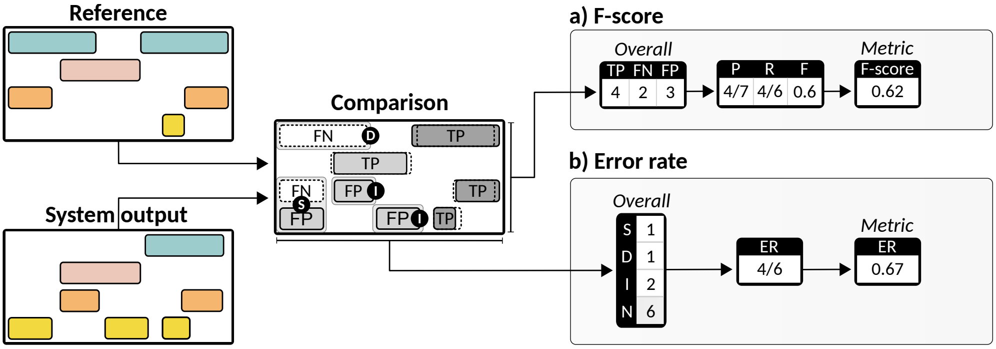


Figure 4.6 Calculation of two event-based metrics: F-score and error rate. F-score is calculated based on overall intermediate statistics counts. Error rate counts total number of errors of different types (substitutions, insertion and deletion).

calculated based on total counts (instance-based, micro-average) or based on class-wise performance (class-based, macro-average). The metric calculation is illustrated in Figure 4.6 panel (a). The event-based error rate is defined with respect to the number of reference sound event instances. The substitutions are defined differently than in segment-based error rate: events with the correct temporal position but incorrect class labels are counted as substitutions, whereas, insertions and deletions are assigned for events unaccounted for as correct or substituted in system output or reference. The overall metric is calculated based on these error counts similarly to segment-based metric with Equation 4.3. The metric calculation is illustrated in Figure 4.6 panel (b).

In comparison to the segment-based metrics, the event-based metrics will usually give lower performance estimate values because it is generally more complicated to match onsets and offsets than overall activity of the event. The event-based metrics measure the ability of the system to detect the correct event in the right temporal position, acting as a measure of onset/offset detection capability. Thus, event-based metrics are the recommended choice for applications where the detection of onsets and offsets of sounds is an essential feature.

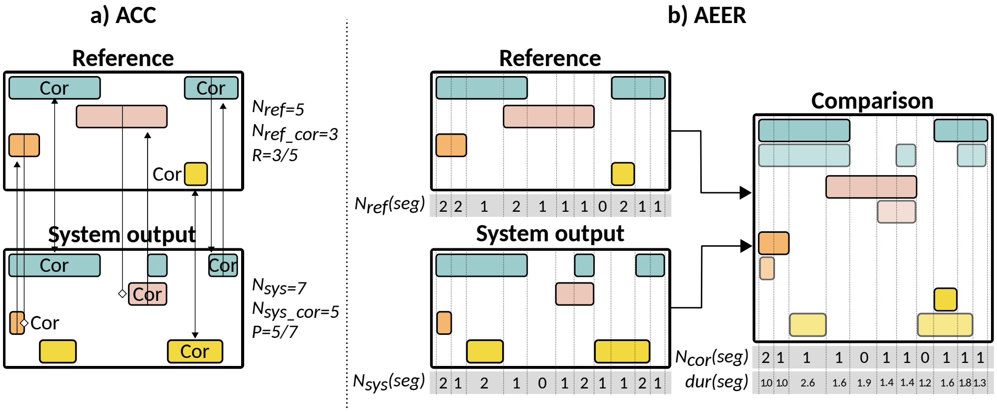

a) ACC
Reference
Cor
Cor
Cor
N_{ref}=5
N_{ref_cor}=3
R=3/5
System output
Cor
Cor
Cor
Cor
Cor
N_{sys}=7
N_{sys_cor}=5
P=5/7
b) AEER
Reference
$N_{ref}(seg)$ 2 2 1 2 1 1 1 0 2 1 1
System output
$N_{sys}(seg)$ 2 1 2 1 0 1 2 1 1 2 1
Comparison
$N_{cor}(seg)$ 2 1 1 1 0 1 1 0 1 1 1
dur(seg) 1.0 1.0 2.6 1.6 1.9 1.4 1.4 1.2 1.6 1.8 1.3

Figure 4.7 Calculation of intermediate statistics for two legacy metrics: ACC and AEER.

4.3.3 Legacy Metrics

Earlier works included in this thesis, [P1] and [P2], used metrics defined for the CLEAR 2006 and CLEAR 2007 evaluation campaigns [174]. These metrics were evaluated only for known non-speech events, and therefore “speech” and “unknown” events were excluded from the calculations. The first metric originating from CLEAR campaign was defined as a balanced F-score and denoted by ACC. In the evaluation, the outputted sound event was considered to be correctly detected if the temporal center of the event lies between the timestamps of a reference event with the same event class, or if there exists at least one reference event with the same event class whose temporal center lies between the timestamps of the outputted event. Conversely, the reference event was considered to be correctly detected if there was at least one outputted event whose temporal center is situated between the timestamps of reference sound event from the same event class, or if the temporal center of the reference event lies between the timestamps of at least one outputted event from the same event class. The calculation process of the intermediate statistics for the metric is illustrated in Figure 4.7 panel a. The metric was defined as

$$ACC = \frac{2 \cdot P \cdot R}{P + R} \tag{4.4}$$

with the precision P and the recall R defined as

$$P = \frac{N_{sys\_cor}}{N_{ref}}, \quad R = \frac{N_{ref\_cor}}{N_{sys}} \tag{4.5}$$

The second metric originating from CLEAR campaign considered the temporal resolution of the outputted sound events by using a metric adapted from a speaker diarization task. The metric defined the acoustic event error rate (AEER) expressed as time percentage [174]. The metric computes intermediate statistics in adjacent time segments defined by onsets and offsets of the reference and system output events. In each segment (seg), the number of the events is counted (N_{ref} and N_{sys}) along with the correctly outputted events N_{cor}. The intermediate statistic calculation for AEER is illustrated in Figure 4.7 panel b. The overall AEER score is calculated as the fraction of the time that is not attributed correctly to a sound event:

$$AEER = \frac{\sum\limits_{seg} \left\{ \mathrm{dur}(seg) \cdot \max(N_{\mathrm{ref}}, N_{\mathrm{sys}}) - N_{\mathrm{cor}} \right\}}{\sum\limits_{seg} \{ \mathrm{dur}(seg) \cdot N_{\mathrm{ref}} \}} \tag{4.6}$$

where $\mathrm{dur}(seg)$ is the duration of the segment.

The ACC metric can be seen as an event-based F-score where the correctness of the events is defined using centers of events instead of their onsets and offsets. Similarly, the AEER metric can be seen as a segment-based metric calculated in non-constant sized segments. As a result, these metrics are measuring different aspects of performance at different time-scales. This is problematic as neither of them will give sufficient performance measures alone, but their simultaneous usage is not advisable due to their different definition. These shortcomings are alleviated by the new metrics (F-score and ER) for segment-based and event-based measurement defined in [113]. These metrics were defined using the same temporal resolution, and even though individually they still provide only an incomplete view of the system performance, they can be used together easily to get a more complete view of the performance. Moreover, the evaluation with AEER is based on non-constant segment lengths determined by the combination of reference events and system output events, making the evaluation segments different from system to system. Error rate defined in [113] uses uniform segment length and simple rules to determine the correctness of the system output per segment, making the metric easier to understand.

4.4 Training Acoustic Models from Audio Mixtures

In everyday environments, sound events can be active simultaneously (overlapping in time and frequency). When captured with a microphone, the sounds are summed up to form an additive audio mixture. The mixture signals with overlapping target sounds pose a difficulty for robust acoustic model learning, and for this reason

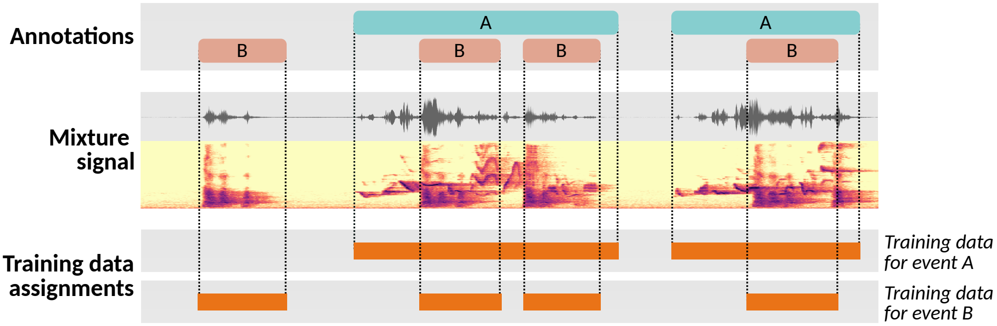

Annotations
A
A
B
B
B
B
Mixture signal
Training data assignments
Training data for event A
Training data for event B

Figure 4.8 Assigning audio segments for learning from material containing overlapping sound events.

often in traditional approaches only isolated and non-overlapping sound events were used as learning examples. For these approaches, isolated sound events are usually collected separately with a minimal amount of interfering sounds, as the everyday environments are usually too complex to provide enough non-overlapping material for reliable model learning. As the tolerance against the interfering sounds is the key for robust detection, the aim should be to use real mixture signals captured in everyday environments also in the learning stage of the development. The works included in the thesis use mixture signal for learning the acoustic models, and deal with the mixtures in two different ways: to assign segments of the mixture signal to multiple target classes (used in [P1], [P2], [P6] and [P7]), or to separate sounds from the mixture signal and assign them to target classes (used in [P3] and [P4]).

4.4.1 Segment Assignment from Mixture signals

The straightforward approach to utilize the mixture signals is to use each annotated event instance for learning regardless of whether there were overlapping events present or not. The segments of the mixture signal that contain overlapping events are used as learning instances during the model learning process for all event classes active within that segment. The basic procedure of assigning the learning data for the sound event classes is shown in Figure 4.8. The assumption behind this procedure is that the variability caused by overlapping sounds in the segments can be seen as noise, and the most dominant shared characteristics are caused by the presence of the target sound class. The noise caused by the overlapping sounds will be averaged out in the model learning given there is a sufficiently large and diverse collection of learning examples available, and the model will learn a reliable representation of the target sound event. This approach has been used in [P1], [P2], [P6] and [P7].

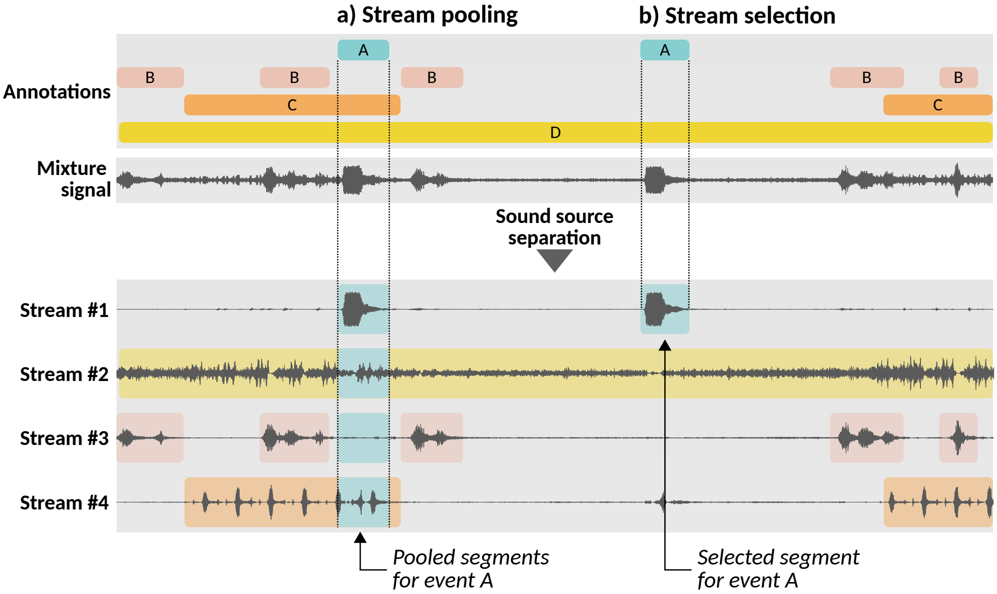


Figure 4.9 Overview of two methods assigning learning instance for event A from four separated audio streams. In stream pooling, all corresponding segments in each stream are assigned for learning, whereas in the stream selection only the most appropriate segment is selected among the streams.

4.4.2 Segment Assignment from Separated Signals

To minimize the effect of interfering sounds in the mixture signal, sound source separation techniques, such as non-negative matrix factorization, can be used to decompose the signal into streams containing individual sound sources. In this approach, the learning examples are pre-processed using NMF by separating them into multiple streams of audio, and the annotations are used to provide the segmentation into event instances. Ideally, a sound source in the audio mixture will be separated into a single audio stream, and overlapping sounds will be separated into other streams. The NMF separation process is unsupervised, and there is no knowledge of which streams contains what event from the annotations. In addition, the expected number of overlapping sound events is unknown, and consequently, the optimum number of streams for the sound source separation requires prior knowledge or assumptions about the complexity of the analyzed data. In [P3] and [P4], number of streams was fixed to four in agreement with the average sound event polyphony of the used audio dataset. The NMF based sound separation was discussed in Section 3.3.1. Two methods of using the separated audio streams in the learning process were studied: *pooling* audio material from all streams together, and *selecting* the most appropriate learning material from the streams. The two methods are illustrated in Figure 4.9.

Pooling Training Material

The approach proposed in [P3] pools all separated streams together for the model training. The segments determined by the annotation of a sound event instance in all the separated streams are assigned to train the annotated class. The approach is based on assumption that the characteristics of the target sound are more dominant in a segment of the separated stream than in the same segment of the audio mixture. The overlapping sounds present in other streams collected during the pooling process will cause some interference and variability to the training material as in the approach utilizing directly segments of the mixture signal. However, as the target sound characteristics are emphasized in one of the separated streams, the model can learn a more robust representation of the target sound events.

Selecting Training Material

In the ideal sound source separation process, one sound source instance is separated into a single stream. By selecting only the corresponding segment from this single stream for training, one should get the most appropriate learning example and minimize the noise introduced by the overlapping sounds. In [P4], the most representative training material was selected iteratively from the separated audio streams. Initial models for the event classes are learned from pooled segments from all the streams based on prior knowledge about the temporal location of events given by annotations. Two approaches to select the stream that contains the target sound were studied: one selecting the most likely stream and another gradually eliminating the most unlikely streams from the training until only one stream is left. The stream selection is based on the expectation-maximization (EM) algorithm [35] which is used to refine the training material during the acoustic model training. In [P4], the method was applied for SED system using sound event class specific HMM models. The stream selection procedure is independent for each class, and thus the process can be presented for only a single event class model λ.

The process is formalized as follows. An audio segment extracted from an annotated time-segment s in a stream with index m is denoted as $x_{s,m}$, while a set of events that are annotated to contain target class is denoted by $\mathscr{C}$. The EM algorithm is then used to associate subset of the $x_{s,m}$ for training acoustic model λ. The initial model is trained with all annotated time-segments S from all four separated streams. These streams are denoted with $x_{\mathscr{C},1:4}$, where all the x are indexed by event set $s \in \mathscr{C}$ and $m \in [1,4]$. Even though the initial model can be considered noisy, containing interfering material from other than the target event class, it is good enough to give sufficient initial guess for the selection process. After this, the EM algorithm operates iteratively by repeating the E step (expectation) and M step (maximization), and at each iteration the likelihood function $P\left(\lambda \mid x_{\mathscr{C},1:4}\right)$ is maximized. Using Bayesian expansion, the expression to be maximized is defined as

$$P\left(x_{\mathscr{C},1:4} \mid \lambda\right) \equiv \sum_{s\in\mathscr{C}} \sum_{m} P\left(x_{s,m}, a_s = m \mid \lambda\right), \tag{4.7}$$

where the latent variable a_s is used for the index of the stream that contains the target event. This equation can be further expanded into

$$P\left(x_{\mathscr{C},1:4} \mid \lambda\right) = \sum_{s\in\mathscr{C}} \sum_{m} P\left(x_{s,m} \mid \lambda\right) P\left(a_s = m \mid x_{s,m}, \lambda\right). \tag{4.8}$$

where $P\left(x_{s,m} \mid \lambda\right)$ is the likelihood of $x_{s,m}$ for the event model. The EM algorithm then iterates over the expectation step by calculating posterior probability, $P\left(a_s = m \mid x_{s,m}, \lambda\right)$ denoted by $a_{s,m}$, and the maximization step by retraining the new model λ:

$$\text{(E):} \qquad a_{s,m} = P\left(a_s = m \mid x_{s,m}, \lambda\right) \tag{4.9}$$

$$\text{(M):} \qquad \lambda \leftarrow \arg\max_{\lambda} \sum_{s\in\mathscr{C}} \sum_{m} P\left(x_{s,m} \mid \lambda\right) a_{s,m}. \tag{4.10}$$

The expectation step in Eq. 4.9 represents the stream selection, and is given as

$$a_{s,m} = \frac{P\left(x_{s,m} \mid \lambda\right)}{\sum_{m'} P\left(x_{s,m'} \mid \lambda\right)}. \tag{4.11}$$

The maximization step in Eq. 4.10 represents the training of the new model for the event given the new selection of training material. The maximization step is simplified by making selection a binary, and using only $x_{s,m}$ for which $a_{s,m} = 1$. This simplification allows using the conventional Baum-Welch algorithm to train a new HMM model without weighted observations. The selection a is made binary based on the stream selection approach either by selecting the most likely stream or by eliminating the least likely stream at each iteration. In the *prominent stream selection*, $a_{s,m}$ having the highest likelihood among $a_{s,1:4}$ is set to one and $a_{s,m'}$ for other m is set to zero. In the *stream elimination*, the n smallest $a_{s,m}$ among $a_{s,1:4}$ are set to zero, i.e., eliminated, where n is set equal to the iteration count. The stream selection of a single learning instance using the two selection approaches is illustrated in Figure 4.10.

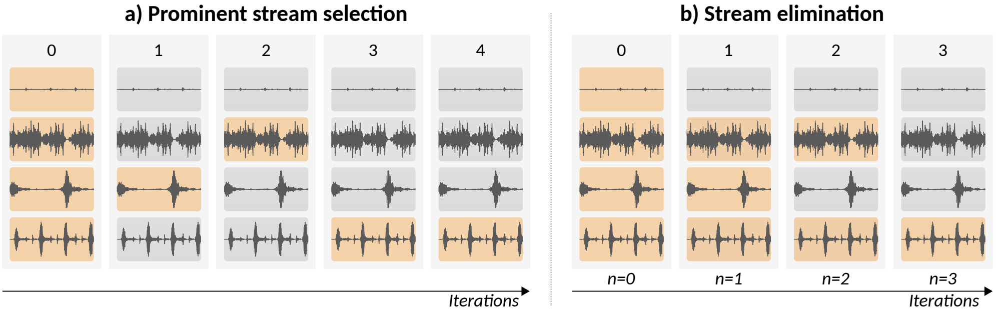


Figure 4.10 Illustrative example of the stream selection using two approaches. Prominent stream selection: in each iteration only one $a_{s,m}$ is set to one, rest are zero. Stream elimination: in each iteration one more $a_{s,m}$ is set to zero.

4.5 Monophonic Detection

Publication [P1] studied monophonic sound event detection for a wide range of everyday environments with a large set of sound event classes. This was the first publication to evaluate SED with manually annotated real-life recordings from a large set of different acoustic scenes. The work was connected to the research from the CLEAR evaluation campaign by using the same evaluation metrics. In the context of this thesis, the publication was a preliminary study on the feasibility of the HMM-based approach for SED: first, the proposed approach was tested in a controlled classification setup to find the best model topology and parameters, and after this, the approach was extended to perform monophonic SED. In publication [P2], the approach was further extended with a universal background model in order to be able to capture sound events unknown to the system. The proposed monophonic detection approach was expanded in the subsequent publications [P2]–[P4] for polyphonic detection.

4.5.1 System Structure

In the system, the coarse shape of the power spectrum was represented using MFCCs, and class-specific feature distributions and temporal dynamics were modeled with HMMs. Class-wise HMMs were first trained separately, and then the acoustic model for the detection was constructed from these models by connecting them into a single network having transitions from each model to any other. The event sequence for the recording was obtained by finding the most probable state sequence through this compound acoustic model using Viterbi decoding. The overview of the system is presented in Figure 4.11. The overall approach is similar to the author's early work in [73].

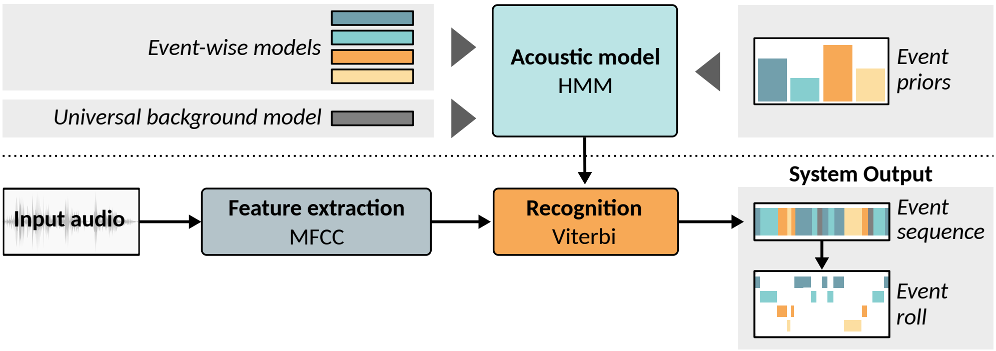


Figure 4.11 Monophonic detection system overview.

Acoustic Features

MFCCs were selected as acoustic features due to their compact representation of magnitude spectra and decorrelated values. The decorrelated values enabled the usage of diagonal covariance matrices in Gaussian distributions used to model the state output distributions of the HMM (see Section 3.3.2). Features were extracted in 20 ms Hamming-windowed analysis frames with 50% overalap, using 40 mel bands, and the first 16 coefficients were retained. A relatively short analysis frame size was selected as the spectral characteristics of sound events are changing rapidly and a good temporal resolution is required in order to capture onset and offsets in the detection. The temporal evolution of the acoustic features was incorporated into the feature vector by including first and second time derivatives of the static coefficients. In order to make the representation less sensitive to the signal amplitude, the zeroth order static coefficient was discarded from the feature vector.

Acoustic Model

Continuous-density HMMs were used to model the sound-event-conditional feature distributions. Based on the experiments, a left-to-right model topology having three states was chosen. States in this topology can be seen to represent a beginning of the sound event (onset), a sustained part of the sound (main body), and an end of the event (offset) in a similar fashion as used to represent musical notes in musical information retrieval and phonemes in speech recognition. The topology provides the flexibility needed to model sound events having naturally varying temporal structures. For example, onsets and offsets can be short for transient-like sounds (e.g. “footsteps”), while the length of the sustained part can accommodate variability in the length of the sound (e.g. “car horn”). Even though this assumption does not hold for all sound event classes, there are clearly some events benefiting from such a temporal modeling. The probability density functions of observations in each state were modeled using a mixture of multivariate Gaussian density functions (GMMs).

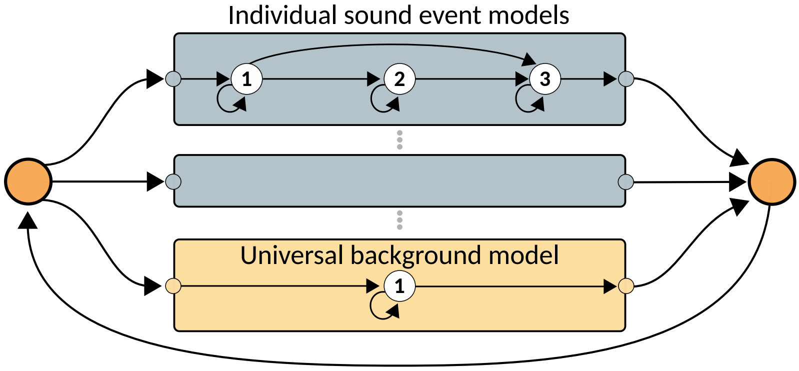


Figure 4.12 Fully-connected sound event model network where sound events are modeled with three-state left-to-right HMMs, and a universal background model with single state HMM model.

Each class-specific HMM was first trained using the EM algorithm (see Section 3.4.3) using time segments from the mixture signals in the training material where the sound event was annotated to be active. This scheme of selecting segments from the mixture signal was described in Section 4.4.1. In order to allow the system to cope with sound events not present in the training material, a one-state HMM trained with all training material was added to capture the general properties of the acoustic scene. This type of model is called a *universal background model* [148]. The acoustic model for the detection was created from the class-specific models and UBM by connecting them into a single network. This network was fully-connected, the output of each model being connected to the input of each model in the network. The resulting network is illustrated in Figure 4.12.

In the sound model network, inter-model transition probabilities are controlled by the prior probability of sound events. Equally probable sound events would have uniform inter-model transition probabilities, and in this case, the network would output an unrestricted sequence of relevant classes, i.e., any event can follow any other. However, in everyday environments sound events are not uniformly distributed, some events are more common than others. For example, "speech" events are much more common than "sneezing" events. The training material can be used to model the prior probabilities for the events, for example by counting event occurrences per class.

Detection

In the detection stage, the output of the detection is a segmentation of the audio signal into regions containing the most prominent event at a time, obtained using Viterbi decoding and the acoustic model. The system output contains the onset and offset times of the recognized prominent events, marked as the timestamps when the search path transitions from one event model to another. Transitions between sound event classes in the acoustic model were controlled by the event prior probabilities acquired from the training material. The balance between prior probabilities and likelihoods given by the acoustic model when calculating the path cost through the model network was adjusted with a weight parameter. In speech recognition, this parameter is referred to as *language model weight*. The number of events in the outputted sequence was controlled using an insertion penalty parameter which adjusts the cost of inter-event transitions. These two parameters were chosen experimentally using a development dataset.

4.5.2 Acoustic Model Hyperparameters

In publication [P1], the hyperparameters of the acoustic model were selected by using sound event classification as a proxy task in two different experiments. The classification task was used instead of detection as it was simpler to evaluate and the conclusions were easier to draw. The aim of the classification experiments was to select the model topology (fully-connected versus left-to-right topology), the number of states in the model, and the number of Gaussians per state. In the first experiment, the Sound Effects 2009 dataset containing isolated sound events was used together with varying environmental noise in different signal-to-noise ratios. In the second experiment, a similar classification experiment was implemented using real-life recordings from TUT-SED 2009 dataset by classifying the most prominent sound event in varying sized segments.

Isolated Sound Events

The first experiment used Sound Effects 2009 dataset with 1359 isolated sound event examples from 61 classes (see Section 4.2) in a five fold cross-validation setup. Various model parameters were first evaluated using isolated sound events with minimal interfering noise, and the best performing setup was then evaluated with varying level of interfering noise. The classification accuracy for various acoustic model parameter combinations are shown in Figure 4.13 panel a. There was quite similar performance across the systems, especially when comparing systems with an equal number of Gaussians. For example, GMM-based system with $nc = 12$ Gaussians, two-state HMM with $nc = 6$, and three-state HMM with $nc = 4$ all produce around 50% accuracy (these data points are visualized in Figure 4.13 with dots). The chosen HMM topology did not have much effect as the state transition probability matrix for the fully-connected models became strongly diagonalized during the training, effectively transforming them into left-to-right models. In the additional experiments, the number of states was adjusted per class according to the average length of sound events,

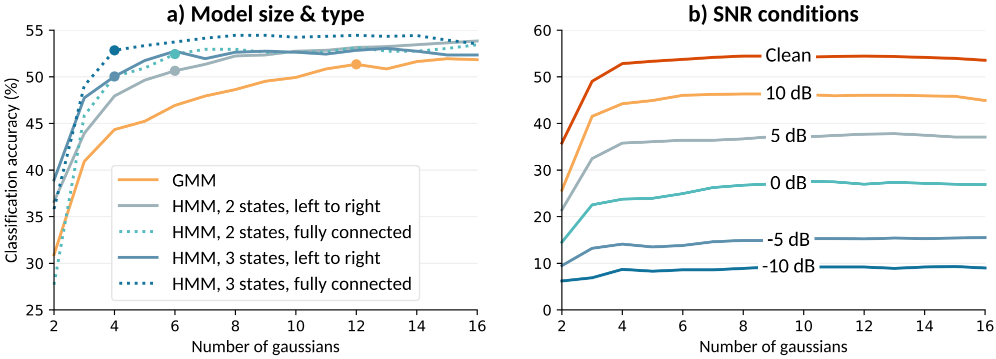


Figure 4.13 Isolated sound event classification accuracy for different acoustic model setups (panel a) and signal-to-noise conditions (panel b). In panel a, points with 12 total Gaussians per class-wise model are indicated for each setup for comparison.

but this did not yield overall improvement in performance. As many parameter combinations provided quite similar performance, HMM-based approach using a three-stage left-to-right topology for all classes was chosen for the later experiments. The decision was made under the assumption that the HMM's temporal modeling capability would be beneficial in sound event detection, and that the chosen topology would benefit modelling of sound event structure with states for onset, sustain and offset parts of the sound for at least some event classes.

In the second experiment, the influence of the environment complexity was studied under controlled signal-to-noise conditions by adding ambient noise to the sound event samples with different signal-to-noise ratios. In this case, the isolated sound event was regarded as the target signal and the noise was an ambience audio example selected from a separate dataset containing background recordings. The ambience sample was randomly selected from the same acoustic scene class as the isolated sound event sample it was mixed with, and the same ambience sample was used when evaluating with different SNR values. The results for three-state HMM with varying number of Gaussians per state and varying SNR are shown in Figure 4.13 panel b. As expected, the classification accuracy decreases considerably with introduced signal complexity. In the later experiments, 16 Gaussians per state were used, in order to ensure sufficient modeling capability.

Sound Events in Everyday Environments

The next experiment studied the sound event classification performance under uncontrolled SNR conditions in real noise conditions of everyday environments. The TUT-SED 2009 dataset was used in five fold cross-validation setup, and the classification was done for each annotated event segment defined by the onset and offset

Table 4.2 Sound event classification results.

	Classification accuracy		
System description	Top-1	Top-2	Top-3
Controlled SNR conditions, Sound Effect 2009 dataset			
Clean	54.5%		
SNR = 10 dB	46.3%		
SNR = 5 dB	37.8%		
SNR = 0 dB	27.6%		
Uncontrolled SNR conditions, TUT-SED 2009 dataset			
Everyday environments	23.8%	35.4%	44.1%

timestamps. The setup was essentially similar to the isolated sound event classification in the first experiment, but the evaluation was done under uncontrolled SNR conditions with naturally polyphonic audio signals where other events could be active within the segment.

The results for experiments with the controlled and uncontrolled SNR conditions are presented in Table 4.2. For classifying recordings from everyday environments, the average classification accuracy over the folds was 23.8%. Since the evaluated segment could contain multiple overlapping sound events, also top-2 and top-3 classification schemes were evaluated. In this scheme, the segment was considered to be correctly classified, if a correct class was among two or three most likely classes. Evaluation results with these schemes showed a good increase in performance, and these observations lead later to the development of a multi-Viterbi based approach for the polyphonic SED [P2].

Under the controlled SNR conditions, the classification accuracy dropped approximately 10% for every 5 dB. At the 0 dB level, when the sound event and the background ambience were at the same level, the classification accuracies are comparable between controlled and uncontrolled SNR conditions. This suggests that 0 dB SNR is on average the approximate level at which annotators can perceive sound events reliably during the annotation process. The class-wise performance for the experiment is shown in Figure 4.14. The majority of the classes had a satisfactory performance level, but at the same time there were also ten classes with close to zero performance.

4.5.3 Results

In publication [P1], the system without UBM was evaluated in a *context-independent* setting where a single global acoustic model was used for all material. In [P2], the

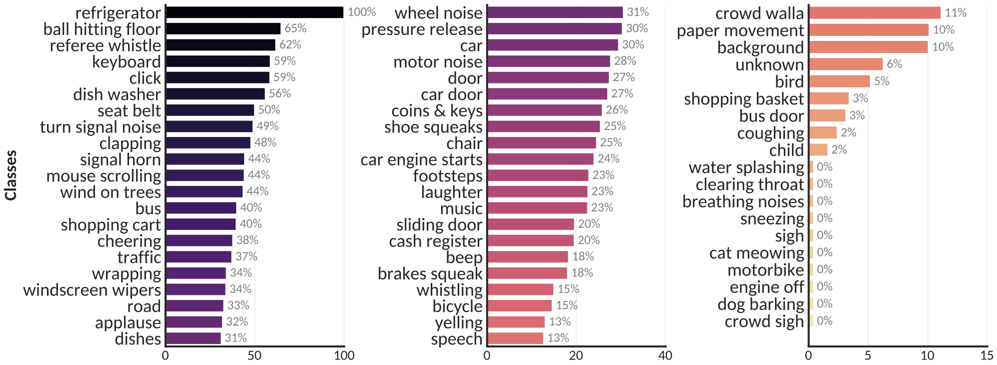


Figure 4.14 Class-wise sound event classification results for uncontrolled SNR conditions.

system was extended with UBM and *context-dependent* approach where the recognized acoustic scene class was used to select a context-specific acoustic model and event priors (see Section 4.7.2). The context-dependent modeling was found to be more effective, and thus later studies on polyphonic SED utilized this approach ([P3] and [P4]).

The TUT-SED 2009 dataset was used in the evaluations. The overall detection results for monophonic SED systems from [P1] and [P2] are presented in Table 4.3. The systems were evaluated using the metrics presented in Section 4.3, which include the CLEAR metrics ACC and AEER, and the proposed segment-based F-score metrics $F_{seg,1sec}$ and $F_{seg,30sec}$. Results for segment-based F-score metrics are presented here to enable easier comparison between monophonic SED results and results presented later in Section 4.6 for polyphonic SED. The best AEER score for the 61 event classes was 84.1% and ACC was 30.1%. These results were comparable to the best performing system in the CLEAR evaluation campaign (36.3% ACC and 99.5% AEER) [162], but the proposed system was capable of detecting a much higher number of event classes (61 versus 12) from multiple acoustic scenes (10 versus 1) with lower AEER score and only slightly lower ACC score. The usage of event priors did not have much effect to the detection performance in the context-independent setting, but in the context-dependent setting a clear improvement can be observed from the use of priors. In the context-independent setting, event priors were calculated over all scene classes and as the “speech” event is the most common event it dominates the prior probabilities and overlaps practically all other events, leading to many confusions in the detection.

The systems using the UBM model had a slightly lower overall performance than the system not using it due to the mismatch between monophonic detection output and polyphonic reference annotations. The outputted UBM event were regarded as silence, and they were skipped in the performance evaluations. In the reference

Table 4.3 Monophonic sound event detection results for systems proposed in [P1] and [P2].

System description	Event priors	ACC	AEER	$F_{seg,1sec}$	$F_{seg,30sec}$	Publication
Systems without universal background model						
Context-independent	Uniform	30.1	84.1			[P1]
Context-independent	Count-based	30.0	84.0			[P1]
Systems with universal background model						
Context-independent	Uniform	28.3	87.0	8.4	17.8	[P2]
Context-dependent	Uniform	33.8	87.8	10.9	27.0	[P2]
Context-dependent	Count-based	40.1	84.2	14.6	29.8	[P2]

annotations, there were not many segments without active events (silent segments) to require UBM event as output. At the same time, some events, especially the quiet ones, might get falsely detected as UBM. The benefit of having UBM in the system was evident when the system was extended to polyphonic detection by using multiple Viterbi iterations [P2]. In this case, the UBM events balanced the temporal change of the polyphony in the individual decoded sequences when active sound events were already detected in other sequences.

4.6 Polyphonic Detection

A system capable of polyphonic detection is essential for well-performing detection in complex everyday environments. The works included in this thesis proposed polyphonic detection approaches in two setups: one that extends the HMM-based monophonic SED system into polyphonic detection using various techniques [P2]–[P4] and another one that uses class-wise activity detectors to detect independently the overlapping sounds [P6]. All presented works use context-dependent system design.

4.6.1 Extending HMM-Based Detection

The monophonic SED system based on HMMs was proposed in [P1]. The subsequent publications extended this approach into polyphonic detection by using multiple Viterbi iterations, multi-Viterbi approach in [P2], or by using sound source separation as a pre-processing step in [P3] and [P4]. These are not truly multi-label extensions, as the systems do not output multiple class labels for the same time instance like the latest approaches based on discriminative classifiers such as neural networks would. Instead, they used a collection of generative classifiers (HMMs) to produce multiple labels for the same time instance.

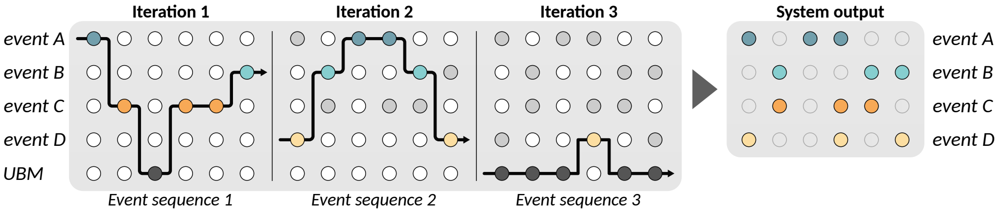


Figure 4.15 Concept of multiple path decoding using three consecutive passes of Viterbi algorithm.

Multi-Viterbi Approach

In [P2] the HMM-based SED approach was extended for polyphonic detection by using multiple consecutive passes of the Viterbi decoding. The approach was adapted from music transcription where it was used for the detection of overlapping musical notes [152]. The main idea of polyphonic detection approach was to iteratively process the test recordings in the detection stage to find the next-best event sequences that are temporally disjoint, i.e., active event classes at each time instance were different. This type of output is difficult to achieve with conventional N-best decoding, because the N-best decoding method provides many paths with only minor state changes between them.

In the detection process, the most likely event sequence was first found. At each next next iteration, time-variant restrictions were imposed to control states the path could enter based on previous iterations. In each time frame, states belonging to the sound event decoded during the previous Viterbi passes were prohibited. UBM events were not restricted, meaning that the system was allowed to use UBM at any time frame in any iteration. This allowed the system to dynamically react to varying levels of polyphony. The Viterbi passes can be iterated until the system selects UBM for each time frame, but in practice the number of iterations can be determined based on the average expected polyphony of the data. The number of iterations gives the maximum number of overlapping events the system is capable to detect. In [P2], the number of iterations was set to four based on the average polyphony of the training material. The example of detection process with the multi-Viterbi approach is shown in Figure 4.15.

Sound Source Separation as Pre-Processing

Publication [P3] proposed a polyphonic extension to monophonic SED system from [P1] in which the input signal was first pre-processed using unsupervised non-negative matrix factorization (see Section 3.3.1). The NMF separates the signal into a number of streams containing roughly homogeneous spectral content that differs significantly from the other streams, essentially separating a combination of the physical sources

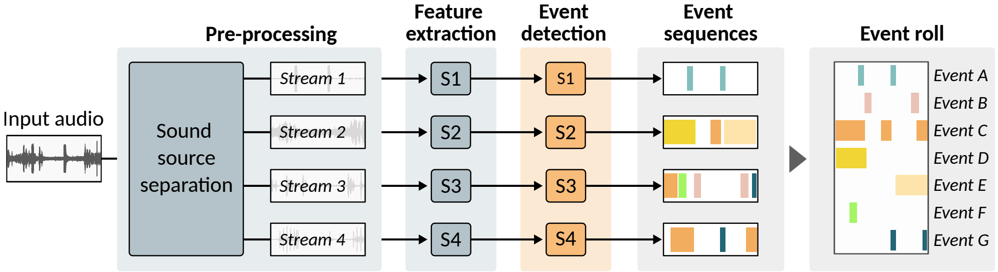


Figure 4.16 System overview of the polyphonic sound event detection using sound source separation as pre-processing method.

into one stream. The aim of this process was to minimize the interfering sounds in a audio stream in order to simplify the detection problem, in consequence making the detection more robust. The monophonic SED system proposed in [P1] was then applied independently to each of the separated audio stream, and detected event sequences were merged across the streams to produce a polyphonic detection output. In the process of merging event sequences across streams, the overlapping events with the same event label were joined into a single event. The number of separated audio stream was set according to the average polyphony of the reference events in the training material. The overall system is presented in Figure 4.16.

In the training stage, the material was pre-processed using the described NMF-based method. The training examples were acquired from these separated signals either using stream pooling [P3] or stream selection [P4] as discussed in Section 4.4.2. The aim of the stream selection scheme was to find most appropriate learning material from the streams, while the stream pooling scheme relied on the assumption that target sound characteristics would dominate material. The universal background model was included in the class set to represent the overall properties of the data and to capture segments in which no events of interest were detected. This model was especially needed in the silent segments of the audio streams, when no sound source was separated into a particular stream.

Results

The proposed polyphonic detection approaches were evaluated using TUT-SED 2009 dataset in a five-fold cross-validation setup using a segment-based F-score metric. The metric was calculated using two segment lengths: one-second segment length ($F_{seg,1sec}$) was chosen to estimate performance for applications requiring a good temporal resolution and 30-second segment length ($F_{seg,30sec}$) for applications focusing more on finding event activity rather than the exact length or onset and offset of the event. The legacy metrics (ACC and AEER) used for monophonic SED evaluation

Table 4.4 Polyphonic sound event detection results for SED approaches proposed in [P2]–[P4]. The evaluation was done using TUT-SED 2009 dataset, and all systems were context-dependent.

System description	$F_{seg,1sec}$	$F_{seg,30sec}$	Publication
Monophonic SED	14.6	29.8	[P2]
Multi-Viterbi approach	20.4	30.0	[P2]
Sound source separation as pre-processing			
Stream pooling	36.7	57.2	[P3] and [P4]
Prominent stream selection	44.5	60.9	[P4]
Stream elimination	44.9	60.8	[P4]

in earlier work [P1] were omitted from the evaluation as they are not that well-suited for evaluating polyphonic detection. As discussed in Section 4.3, they are complementary metrics that measure different aspects of performance at different time scales and this makes their use complicated when optimizing the polyphonic SED systems. The AEER metric uses non-constant segment length in the evaluation defined by the combination of reference events and system output events. The polyphonic system output is composed by more and smaller time-segments than a monophonic output for the evaluation, making the comparison between the systems challenging.

The results are presented in Table 4.4. All polyphonic systems were able to detect a maximum of four overlapping events. Compared to previous work on monophonic SED [P1], the performance was significantly increased when using proposed approaches for polyphonic SED. This performance increase with the polyphonic output was conceptually expected compared to monophonic. The fact that polyphonic systems were capable of detecting more than one event at time enabled them also to detect correctly more than one event in the case of overlapping events, and this increased their recall significantly. The multi-Viterbi approach increased the performance especially when F-score was measured using one-second segments. At the same time, on more coarse time-resolution, the approach did not have much effect on the estimated performance level. This was somewhat expected at the current relatively low performance level. The fragmented output characteristic for the monophonic system can capture small segments of the overlapping events as they sequentially become prominent in the mixture, enabling the monophonic system to detect them and in consequence perform equally well as the polyphonic one on a long time-scale such as the one taken into account by $F_{seg,30sec}$ metric.

The polyphonic systems using sound source separation as pre-processing increased the performance significantly compared to both monophonic SED and multi-Viterbi approach, providing two to three times higher performance on both evaluated segment

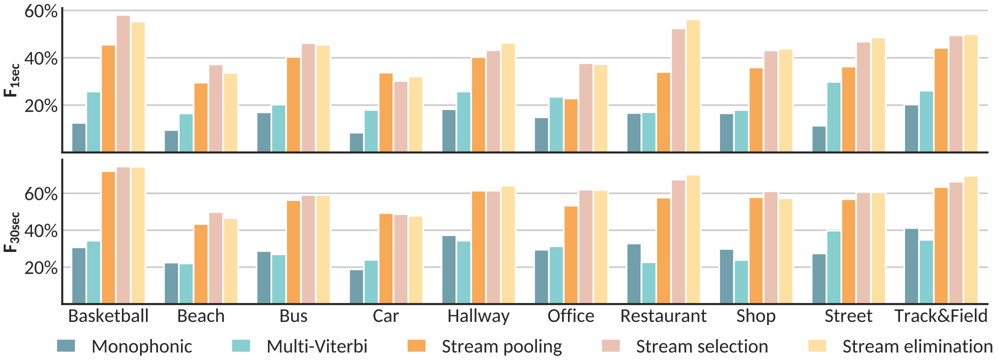


Figure 4.17 Comparison of scene class-wise results from polyphonic sound event detection approaches proposed in [P2] and [P4].

lengths. Stream selection and stream elimination techniques for find appropriate training material selection provided similar performance, increasing the performance compared to the stream pooling technique. In the prominent stream selection, the main difficulty was to determine the number of iterations needed. In the evaluation, the best performance was reached after three iterations, and only minimal changes (under 0.1%) were observed in the performance after that. This technique was found to select mostly the correct streams for training already at the first iteration, as the performance increased only a few percentage units after this. In the stream elimination approach, iterations were performed until only one stream was left. Throughout the iterations performance increased gradually, and reached maximum at the end (after 3 iterations) when only the most likely stream was left in the training material.

The scene-class wise results are presented in Figure 4.17. The most significant performance improvements were observed in the complex environments having many overlapping events, for example, in scene classes such as “basketball game” and “restaurant”. In the noisy acoustic scenes with relatively low amount of overlapping events, for example “inside a car”, the performance increase was more moderate.

4.6.2 Class-wise Activity Detection

Publication [P6] proposed a SED system based on class-wise activity detection. The activity of each event class was detected independently, and class-wise event sequences were then merged to produce the polyphonic system output. The system could be seen as a frame-based sound classifier, where the post-processing of the consecutive frames produces the polyphonic sound event detection output. The system was designed to be simple to train and to implement, and it was meant to be used as the benchmark system for DCASE2016 challenge SED task [P7].

The overview of the system is illustrated in Figure4.18. The activity detectors

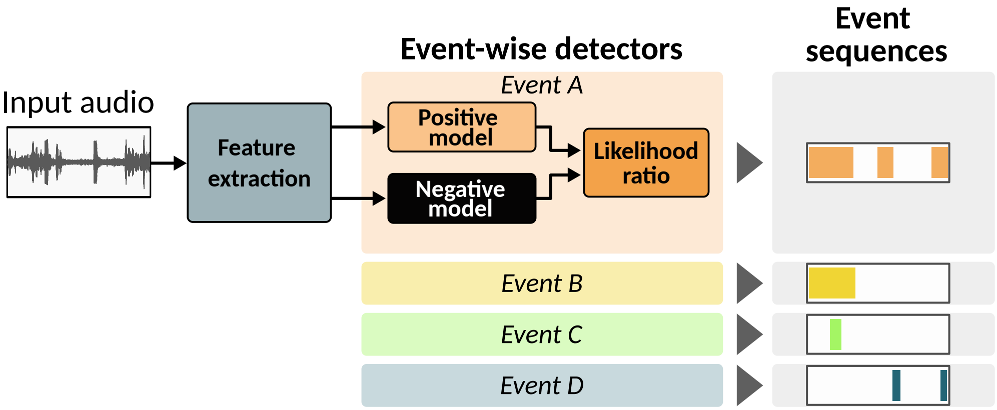


Figure 4.18 Polyphonic sound event detection using class-wise activity detectors.

are based on modeling the activity of the event with one model (positive model) and non-activity with another model (negative model). The approach follows the traditional sound classification approach: MFCC extracted in 40 ms frames were used as acoustic features (20 static, 20 delta, and 20 acceleration coefficients), and GMMs were used as acoustic model (16 Gaussians per model). The positive model was trained using active event segments from the mixture signals according to the annotations, as explained in Section 4.4.1. The negative model was trained using the rest of the training material. The system did not model temporal aspect of the events at all. In the detection stage, a decision on each event activity was taken in each frame. The decision was based on the likelihood ratio between the positive and negative models for the class. The class-wise output was post-processed by taking a majority vote within a one-second sliding window.

Results

The proposed approach was evaluated using TUT-SED 2016 dataset in a four-fold cross-validation setup using segment-based and event-based F-score and error rate metrics. The intermediate statistics from all folds were accumulated to produce a single evaluation result to avoid bias caused by the data imbalance between folds [49]. The segment-based metric was evaluated using a one-second segment length, and the event-based metric used a 200 ms collar for onset condition, 200 ms collar, or tolerance of 50% with respect to reference event length for offset condition. The system was evaluated in a context-dependent setup by training separate acoustic models for both acoustic scenes in dataset. There were 11 event classes in “home” scene, and 7 event classes in “residential area” scene.

Evaluation results are presented in Table 4.5. The segment-based results were fairly good, while on the other hand the event-based performance was quite poor. The poor

Table 4.5 Results for polyphonic SED based on class-wise activity detection from publication [P7].

	Segment-based metrics		Event-based metrics	
Acoustic scene	F-score	Error rate	F-score	Error rate
Home	18.1	0.95	2.5	1.33
Residential area	35.2	0.83	1.6	1.99
Average	26.6	0.89	2.0	1.66

event-based performance is explained by the lack of an explicit segmentation step and lack of temporal information modeling in the training process. In comparison to the other polyphonic detection approaches proposed in this thesis, the class-wise detector approach has a performance level between the multi-Viterbi approach and the approach using sound source separation as pre-processing (according to $F_{seg,1sec}$ metric in Table 4.4) although a direct comparison cannot be made due to the use of different training and evaluation datasets.

4.6.3 Comparison to State-of-the-art

Further development of polyphonic sound event detection systems has brought significant performance improvements. A comparison of the methods presented in this thesis with later proposed approaches is presented in Table 4.6.

The method in Mesaros et al. [108] is based on coupled non-negative matrix factorization. The system learned non-negative dictionaries through joint use of spectrum and class activity annotations, and these dictionaries were then used to estimate the class activities for test signal. In modern standards, the approach is still rather traditional, however, it provided substantial improvement to the detection performance.

Deep learning approaches allowed a straightforward way of modeling polyphony by using multiple neurons in the output layer to model simultaneous activity of multiple sound events. Systems of Çakır et al. [23], Parascandolo et al. [136], and Çakır et al. [24] were context-independent systems capable of discriminating among all 61 classes present in the dataset. The acoustic models for these systems were trained using mixture signals as presented in Section 4.4.1, using mel-band energies as features (see Section 3.3.2). The approach in Çakır et al. [23] used a feedforward neural network with two hidden layers, and the temporal information was incorporated into the network input by concatenating consecutive frames into one feature vector. The system output was processed using median filtering to produce smoother detection results. Parascandolo et al. [136] proposed the use of bi-directional long short-term memory (BLSTM) based recurrent neural networks to model directly the temporal

Table 4.6 Comparison of polyphonic SED approaches included in the thesis with other approaches proposed later. All highlighted works use TUT-SED 2009 dataset and the same cross-validation setup.

	Detection approach	Overlap handling	Context-dependent	$F_{seg,1sec}$
Systems included in this thesis				
Multi-Viterbi [P2]	HMM	At detection	yes	20.4
Stream elimination [P4]	HMM	In pre-processing	yes	44.9
Class-wise activity detection [24]	GMM	System design	yes	34.6
Other systems				
Mesaros et al. [108]	Coupled NMF	Dictionary	yes	57.8
Çakır et al. [23]	DNN	Acoustic model	no	63.0
Parascandolo et al. [136]	BLSTM	Acoustic model	no	64.6
Çakır et al. [24]	CRNN	Acoustic model	no	69.1

information. The approach enabled the system to use information about the past states in order to produce smooth system output without any output post-processing. Later, Çakır et al. [24] introduced a convolutional recurrent neural network architecture for polyphonic sound event detection, combining the convolutional neural network's capability to learn a time and frequency shift-invariant model with the recurrent neural network's capability to model long-term temporal dependencies.

The proposed deep learning approaches had almost twice higher performance in comparison to the classical pattern classification approaches used in [P2]–[P4]. Deep-learning based methods have since become the most popular solution for acoustic modeling, producing state-of-the-art performance in many different tasks related to audio content analysis in the everyday environments.

4.7 Contextual Information

In an everyday environment, sound events occur in relation to other events in the auditory scene and some are specific to certain environments as was discussed in Section 2.2. Co-occurrence of sound events provides *contextual information* that helps identification of both sound events and acoustic scenes. On a short time-scale, co-occurrence can help the identification of one ambiguous sound based on the other identified sounds. On a longer time-scale, identified sound events build up information characteristic to the acoustic scene and help identify the scene. Humans are using contextual information to identify correctly otherwise acoustically similar sound events or acoustic scenes. Sound events are disambiguated based on the contextual information created by the other sound events co-occurring in the scene and additional information about the overall acoustic scene. In many cases, similarly sounding sound events can be disambiguated by identifying first the acoustic scene they are occurring in and based on this information narrowing down the selection of possible sound sources. Acoustic scenes, on the other hand, can be sometimes identified based on the distinctive and scene-specific sound events occurring in the scene.

Computational approaches can use contextual information similarly to human perception in both sound event detection and acoustic scene classification. Some sound event classes are highly unlikely in certain acoustic scenes (see Figure 4.3), therefore information about the acoustic scene can be used to limit the sound event classes used in the detection process and used to select scene-specific acoustic models. This observation enables building a sound event detection system that can handle a large overall set of sound event classes and is easily extendable to new acoustic scenes.On the other hand, the co-occurrence of sound events on a long time-scale gives information about the acoustic scene, and in consequence the acoustic scene class for the recording can be determined based on the sound events detected in it.

Co-occurrence of sound events can also be modeled mathematically into latent topics, using probabilistic latent semantic analysis (PLSA) [112]. Based on the co-occurrence of events represented as the degree of their overlapping in a fixed-length segment of polyphonic audio, the relationships between individual events can be modeled using PLSA. This information can then be used in the detection as a language model, to continuously adjust the probabilities of events according to the history of events detected so far in the acoustic scene. In [112], the event probabilities provided by the PLSA model were integrated into a monophonic sound event detection system proposed in [P1]. Overall this approach provided only a moderate performance increase, but in some specific acoustic scenes, the performance increase was significant.

Language models used in automatic speech recognition provide another approach to modeling relationships between sound events. In this approach, sequences of events are characterized by bigrams, a model of how sound events follow each other. This model could be constructed, for example, based on the annotated onsets in the training material of the system. In comparison to speech, there is a larger set of possible sounds and combinations of sounds, and this, unfortunately, leads to a much more sparse language model. In this case, mechanisms such as model smoothing and backing off to the unigram model have a very strong influence on the model, and the resulting model is very similar to a count-based unigram language model. Therefore in [P2], only count-based unigrams were used as they are easier to create and use.

Table 4.7 Acoustic scene classification systems and their evaluation results.

Publication	[P5]				[P2]			[P6]
System characteristics								
Analysis frame	20 ms							40 ms
Static MFCC coefficients (total feature vector size)	16 (48)							20 (60)
Number of Gaussians	16				32			32
Evaluation results								
Dataset	TUT-SED 2009							TUT-ASC 2016
Scene classes	10							15
Segment size (sec)	4	20	30	40	4	20	40	30
Classification accuracy (%)	70.2	76.4	78.4	80.3	70.0	80.7	85.5	72.5

4.7.1 Acoustic Scene Classification

In the acoustic scene classification, the aim is to classify an audio signal into a predefined scene class that characterizes the environment where the audio signal was captured. Acoustic scene classes can be, for example, “street”, “office”, and “restaurant”. The scene class for the signal can be used as contextual information in the sound event detection as will be discussed in Section 4.7.2. Commonly, the scene classes are modeled based on global acoustic characteristics of the environment. As will be discussed in Section 4.7.3, also sound event activity can be used to model the scene classes.

In publications [P2], [P5] and [P6], a traditional approach modeling global characteristics of the environment was successfully used for the scene classification: MFCCs (static, delta, and acceleration coefficients) were used as acoustic features and the scene dependent distribution of these features was modeled with GMMs. The system parameters and evaluation results of these systems are collected in Table 4.7. The TUT-SED 2009 dataset used in [P2] and [P5] was originally collected with sound event detection in mind and therefore the variability of the location types within scene classes is low, the classification yielding slightly higher performance than for TUT-ASC 2016 dataset that was collected especially for acoustic scene classification and therefore has larger acoustic variability. The results show that acoustic scene can be recognized robustly among a small and restricted set of classes. Furthermore, reasonable accuracy can be obtained already with relatively short classification segment lengths (e.g 4 seconds).

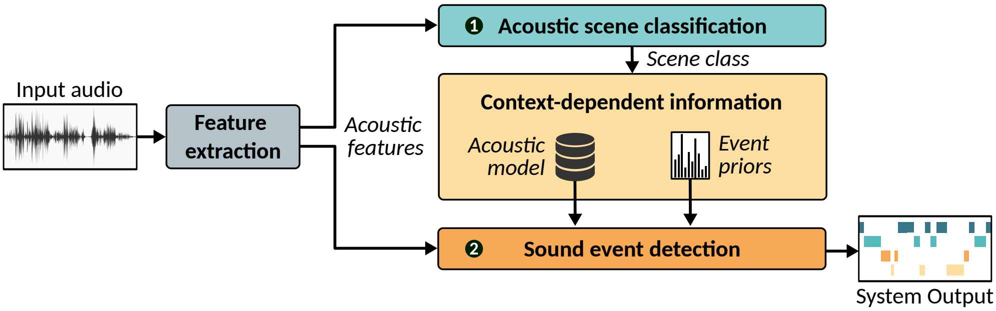


Figure 4.19 Overview of the context-dependent sound event detection system.

4.7.2 Context-Dependent Sound Event Detection

Publication [P2] proposed a *context-dependent* sound event detection system, where the contextual information was used in three ways to help in detection. The overall system is based on the multi-Viterbi approach described in Section 4.6.1 for the polyphonic SED. In the system, the information about the acoustic scene class was used to simplify the detection problem by limiting the selection of used sound event classes. The acoustic modeling is made more robust by training scene-specific models and selecting the used model based on the recognized scene class. The robustness of the detection is further increased by using scene-specific count-based event priors.

The selected approach enables the construction of a SED system capable to operate robustly in a large selection of acoustic scenes and with a large set of sound event classes by using contextual information to simplify the system training and detection processes. The approach also enables an easy system extension to new scene classes without a need to retrain all acoustic models.

System Structure

The system consists of two consecutive steps. In the first step, the acoustic scene class was recognized based on the overall acoustic characteristics of the signal as explained above. In the second step, the recognized scene class was used as contextual information to select the acoustic models and count-based event priors for the sound event detection that is applied for the signal. The overview of the system structure is illustrated in Figure 4.19.

Acoustic scene classification was implemented with a simple but effective approach using MFCCs as acoustic features and GMMs as a classifier (see Section 4.7.1). The audio signal was segmented into non-overlapping 4-second segments, and class-wise log-likelihoods per segment were accumulated over all the segments in the signal. The scene class having the highest total likelihood was outputted as the scene class

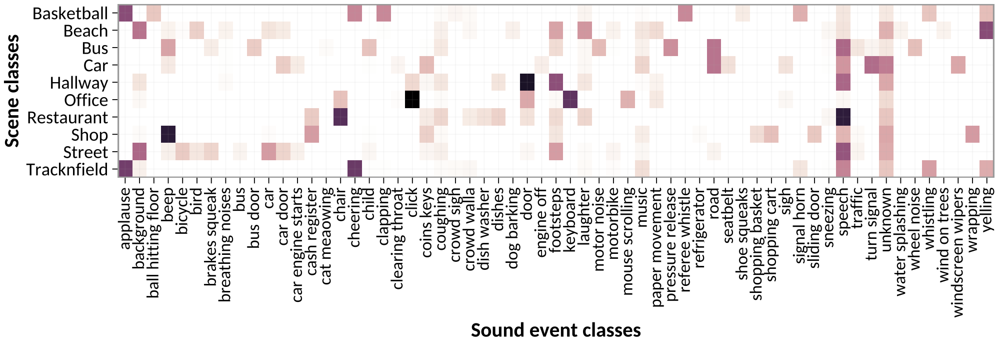


Figure 4.20 Count-based event priors calculated per acoustic scene classes for TUT-SED 2009 dataset.

label for the whole signal. The scene classification will affect the performance of sound event detection, as the incorrect classification will lead to a wrong set of sound event classes, and sub-optimal event priors and acoustic models to be used in the detection. However, a wrongly classified scene class does not necessarily mean entirely incorrect sound event detection output. In this case, the system will miss the scene-specific sound events, but there are always some sound event classes that are present in multiple acoustic scenes and some of these common events will be detected even with a sub-optimal acoustic models.

Sound-event-conditional feature distributions were modeled using continuous-density HMMs using a three state left-to-right model topology. Furthermore, a one-state HMM trained with all training material, universal background model, was included to capture unknown events. This model was essentially the same as used for acoustic scene classification in the previous step. For a more accurate modeling, acoustic models for sound events were trained separately for each acoustic scene using training material only from the specific scene class. Count-based event priors were calculated from the reference annotations of the training material for each acoustic scene class by counting event instances and normalizing with the total event count. The priors for TUT-SED 2009 datasets are illustrated in Figure 4.20. The balance between the event priors, the acoustic model, and the sequence length is controlled by the language model weight and insertion penalty. These were chosen based on a development set by optimizing the system to detect a number of sound events approximately equal to the number of events available in the reference annotations. In the test stage, the set of possible sound events is determined by the scene label provided by the acoustic scene classification step. The scene-specific HMM models of these sound events were connected into a single HMM with transitions from each event model to any other event model (see Figure 4.12) and scene-specific event priors were used to control the transition probability from one event model to another event model.

Table 4.8 Evaluation results for context-dependent sound event detection.

System description	Monophonic system		Polyphonic system	
	$F_{seg,1sec}$	$F_{seg,30sec}$	$F_{seg,1sec}$	$F_{seg,30sec}$
Context-independent system				
Uniform event priors	10.0	21.9		
Count-based event priors	12.0	25.8		
Context-dependent system, oracle scene class				
Uniform event priors	10.9	27.0	19.8	28.9
Count-based event priors	14.8	30.2	20.4	30.0
Context-dependent system, recognized scene class				
Uniform event priors	10.9	27.0	18.9	29.8
Count-based event priors	14.6	29.8	19.5	29.4

Results

In the evaluation, both the acoustic scene classification and the sound event detection steps used similar parameters for features and acoustic model. MFCC features were extracted in 20-ms analysis windows with 50% overlap using 40-channel filter-bank, and the final feature vector consisted of static MFCC coefficients and the first and second-time derivatives. In the acoustic models, the number of Gaussian distributions per state was fixed to 16. The proposed approach was evaluated using TUT-SED 2009 dataset in a five fold cross-validation setup and the evaluation results were averaged over the folds.

Evaluation results are summarized in Table 4.8. Monophonic and polyphonic SED setups were included in the evaluation with uniform and count-based event priors. The number of Viterbi passes was fixed to four based on the average polyphony of the annotated training material, therefore the system was able to detect maximum four overlapping events. In the evaluation, the context-dependent detection substantially improved the performance compared to the context-independent detection approach. This is partly due to the context-dependent event selection simplifying the task, and partly due to more accurate sound event modeling within the acoustic scene being able to represent particular characteristics of the sound event specific to the context. Evaluation using oracle scene class compared to using the recognized scene class shows only minimal difference in the performance metrics even though the acoustic scene classification step introduced 9% error in the scene recognition. Results per scene class, presented in Figure 4.21, show how the context-dependent approach increases the detection performance in all scene classes.

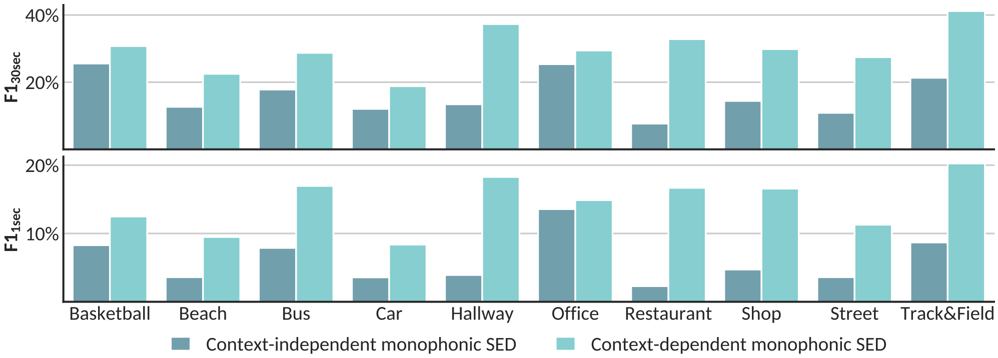


Figure 4.21 Results per scene class for context-independent SED and proposed context-dependent SED systems.

4.7.3 Acoustic Scene Classification Based on Sound Events

Publication [P5] proposed use of long term co-occurrence of sound events as contextual information for acoustic scene classification. Most prior work in acoustic scene classification is based on modeling global acoustic characteristics of the audio signal rather than actual sound events happening in it. The proposed approach assumes that different acoustic scenes, such as a street or a restaurant, are characterized by the occurrence of certain sound events. Acoustic scenes were modeled with *event histograms* generated from manually annotated recordings.

System Structure

The proposed system is divided into two steps: sound event recognition and acoustic scene classification. The audio content analysis system was used to recognize sound events from the audio signal, and the event histogram constructed from the recognition result was then compared with acoustic scene models.

The event histogram presents the number of occurrences per sound event class in the audio signal, omitting information about events order and their lengths. In order to prevent a bias towards longer recordings with more events, occurrence counts in the histogram were normalized by dividing them with the total event count of the recording. The model for each acoustic scene was constructed by summing up event histograms from all training examples from the scene class. The resulting histogram was normalized so that the bins sum up to one. The acoustic scene classification was based on comparing the histogram generated for the test recording with the scene model histograms by calculating a distance between them.

Two alternative schemes to produce the events for the histogram are illustrated in Figure 4.22. In the segment-based scheme, context-independent sound event

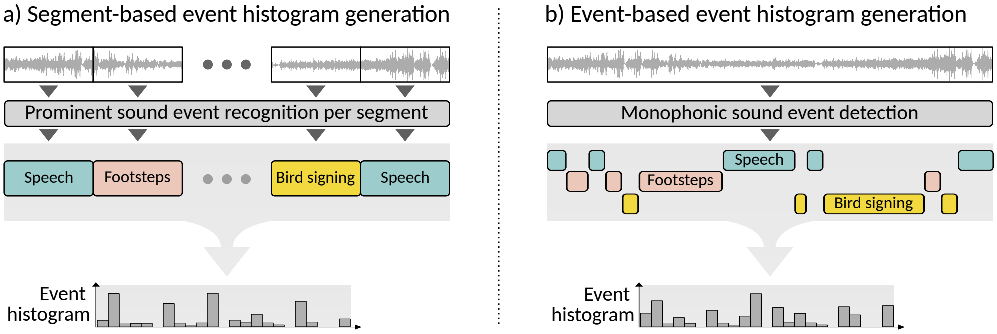

a) Segment-based event histogram generation
Prominent sound event recognition per segment
Speech
Footsteps
Bird signing
Speech
Event histogram
b) Event-based event histogram generation
Monophonic sound event detection
Speech
Footsteps
Bird signing
Event histogram

Figure 4.22 Two schemes to create a event histogram for audio recordings: a segment-based and event-based.

recognition was applied in four-second segments to output the most likely sound event class (most prominent event). In the event-based scheme, context-independent monophonic SED was applied for the whole signal to get the most likely event sequence. These systems were published previously in [P1], and they were trained to recognize 61 sound event classes.

The event histograms were compared using cosine distance, defined as the cosine of the angle between an event histogram of the known scene class C and the event histogram of the tested recording Q:

$$Dist_{cos}(Q,C) = \frac{\sum_{i=1}^{T} q_i c_i}{\sqrt{\sum_{i=1}^{T} q_i^2 \sum_{i=1}^{T} c_i^2}} \tag{4.12}$$

where q_i is the normalized event count of event class i in the tested recording, c_i is the normalized event count of event class i in the scene class and T is number of events in the vector. The scene class corresponding to the lowest distance was selected as the system output. The within-class variation in the distribution of the events can be modeled more accurately by using instance-based scene modelling scheme. In this approach, all training instances (individual recordings) were used to represent the scene class, and the acoustic scene was represented using a set of separate event histograms. The k-nearest neighbor (kNN) approach can be then used for the classification. The distances from the test histogram were first calculated to all histograms from the training material, and after that the classification was done by majority voting among k nearest histograms.

The importance of events in the histogram distance calculation can be controlled using term frequency-inverse document frequency (TF-IDF) weighting approach [149, 175]. When using this approach, the sound events were used as the indexing terms and recordings were seen as documents containing these terms. Depending on the used classification scheme, the document is either an individual recording or all recordings belonging to a scene class. The main idea in this weighting approach is that a term is an important indexing term for document d if it occurs frequently in it, whereas terms occurring in multiple documents are less important for indexing due to their common nature. These two aspects are denoted as term frequency (TF), and inverse document frequency (IDF). The inverse document frequency value for a term is low when the term occurs in many documents and highest when the term occurs in only one. The inverse document frequency is defined as follows:

$$IDF(term) = \log\left(\frac{|D|}{DF(term)}\right), \tag{4.13}$$

where $|D|$ is the total number of recordings and $DF(term)$ is the number of documents in which the term occurs at least once. The weight w_i of a term i in document d is calculated as

$$W_i = TF(term_i, d) \cdot IDF(term_i), \tag{4.14}$$

where $TF(term_i, d)$ is the term frequency, i.e., the number of times $term_i$ occurs in the document d. In the system learning stage, the model histograms were weighted based on the TF and IDF calculated from the training material. In the classification stage, weighting was done by using event histogram of the test recording as TF, while IDF was estimated from the training material.

Results

The proposed acoustic scene classification approach was evaluated using TUT-SED 2009 dataset in a five fold cross-validation setup and the evaluation results were averaged over the folds. In the dataset, there are 103 recordings from 10 scene classes. The evaluation results are presented in Table 4.9. Compared to the baseline system implementing acoustic scene classification based on global acoustic characteristics of scene, the proposed approach produced at best only comparable results. TF-IDF weighting helped the classification only when using instance-based scene models. As the TF-IDF weighs more the rare events than the common ones, having one global scene model for complex scene classes will smooth out the rare events. In addition to this, weighting has problems in case of short recordings having a small number of common events as they would be weighted to zero.

The proposed event histogram-based approach uses complementary information compared to the traditional GMM-based approach discussed in Section 4.7.1. The traditional approach models the general acoustic characteristics of the scenes, whereas the proposed approach models event distributions of the scenes. To take advantage of this complementary information, late fusion scheme was used to merge outputs of

Table 4.9 Acoustic scene classification results for the proposed approach using event histograms. In addition, results for the GMM-based system based on global acoustic characteristics of scene and the system combining it with the proposed (fusion) are presented.

Histogram		Global scene modeling		Instance-based scene modeling				
Generation	Weighting	Single	Fusion	$k=1$	$k=3$	$k=5$	$k=7$	$k=9$
GMM-based system		88.5						
Event histogram-based systems								
Segment-based	No	88.5	91.4	87.3	84.6	85.8	84.8	83.8
Segment-based	TF-IDF	61.1	90.5	89.3	85.6	84.6	85.5	86.6
Event-based	No	84.5	92.4	86.4	84.6	84.6	82.6	81.5
Event-based	TF-IDF	59.3	90.4	89.3	87.5	87.5	89.4	89.4

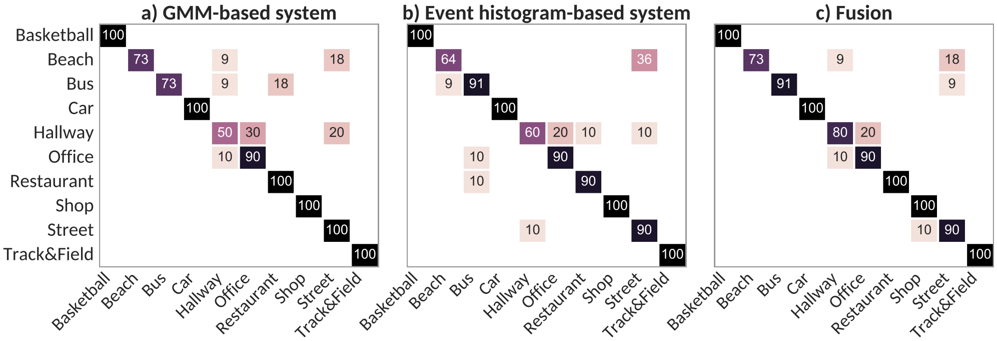


Figure 4.23 Confusion matrices for three acoustic scene classification approaches: GMM-based system, proposed event histogram-based system, and system combining these two approaches using late fusion.

these two systems. The distance values between histograms in the proposed system were mapped into probabilities with inverted sigmoid-function, which were multiplied with the scene likelihoods produced by the GMM-based system. This combined approach provided slightly better accuracy and robustness in the case of acoustically similar scene classes. Confusion matrices for the evaluated systems are shown in Figure 4.23. The proposed event histogram-based system increased the scene-wise performance mainly for bus and hallway scene classes, and the performance increase was preserved after the fusion. For example, in the fusion system the confusions for the “bus” class are made with “street”, which seems more reasonable mistake than “hallway” and “restaurant” as with the GMM-based system.

4.8 DCASE Evaluation Campaigns

In recent years, the DCASE evaluation campaigns have had a substantial role in the growth of the research interest towards the audio content analysis of everyday environments. The *DCASE challenge* presents each year 3-6 tasks, each task focusing on a single target application and research question. Tasks related to acoustic scene classification and sound event detection have been an integral part of each challenge edition organized so far. The DCASE challenge was organized for the first time in 2013 [166], and since 2016 it has been an annual event together with the DCASE workshop organized after each challenge. The challenge attracts increasing amount of participants each year from academic and industrial research teams. For example, challenge in 2020 received 473 submission entries from 138 teams to six challenge tasks, and sound event related tasks received submissions from 35 teams. The DCASE challenge has had an important role in steering the research domain towards open data and reproducibility by establishing standards for evaluation protocols, open source metric implementations, and open benchmark datasets. These factors have supported the growth of the research community and attracted interest also from researchers coming from neighboring research fields.

Some works included in this theses are directly related to organization of DCASE challenge tasks on sound event detection in real-life audio under campaign editions 2016 [P7] and 2017 [109]. This section presents contributions related to the tasks organization which includes task planning [P6], releasing open datasets and open source baseline systems ([P6] and [106]), defining evaluation metrics for the task [113], and analyzing the submitted systems ([P7] and [109]). In the task organization, the work focused on the preparation stage and results analysis stage, and both stages were documented in publications. In the preparation stage, the task was defined, evaluation metrics were selected, dataset was prepared, and the baseline system was evaluated with the dataset. This stage was documented in publication ([P6] and [106]) to provide a clear reference to the task in subsequent publications. In the results analysis stage, organizers evaluated the submitted system outputs and published results on the challenge website. In-depth analysis of the submitted systems and their results were published after each editions as publications [P7] and [109]. Between these stages, participants developed their systems using the development dataset, produced the system output based on the evaluation dataset, and prepared the technical report to describe the system thoroughly.

4.8.1 Sound Event Detection in DCASE Challenge

The DCASE challenge has presented sound event detection in various setups. In editions 2013, 2016, and 2017, the task was set up as a traditional supervised sound event detection problem using recordings with strong annotations. As strong annotations are laborious to produce for a large number of recordings reliably, DCASE challenge has also included SED and SELD tasks using synthetically generated data (2013 and 2016-2020) and using weakly annotated training material (2017-2020).

The sound event detection tasks included in DCASE 2016 and 2017 had a similar task setup: the dataset contained recordings captured in real-life environments which were manually annotated with event labels and event onsets and offsets, and the challenge ranking was based on segment-based error rate calculated in one-second segments. The task setup for DCASE 2016 ASC and SED tasks were described in [P6] along with the dataset information, baseline systems and their performance. For DCASE 2017 challenge, all task setups were described in [106].

Datasets

The DCASE 2016 task used TUT-SED 2016 dataset [116, 120] described in Section 4.2, containing material from two acoustic scene classes: "residential area" was selected to represent outdoor scene typical for surveillance applications, and "home" was selected to represent human activity monitoring or home surveillance applications. The dataset had 18 sound event classes. The development dataset had 1h 18min of audio containing 954 event instances and the evaluation dataset 35min of audio and 511 instances. DCASE 2017 task used TUT-SED 2017 dataset [121, 122] consisting of a single acoustic scene class ("street") that was selected to represent an outdoor environment of interest for detection of events related to human activities and hazard situations. The dataset had 6 sound event classes; the development dataset had 1 h 32 min of audio containing 659 event instances and the evaluation dataset 29 min of audio and 247 instances.

The data in TUT-SED 2016 and TUT-SED 2017 was first partitioned into development and evaluation datasets, and the development dataset was split further into four fold cross-validation setup. The splits were done based on the amount of examples available for each event class, taking into account recording locations. Ideally, the subsets in the split should have the same relative amount of data for each event class, for example, 70% of instances for training and 30% for testing. However, since in real-life environments the event instances are not evenly distributed within the recordings, the splitting condition had to be relaxed. For example for DCASE2016 dataset, the condition was relaxed to include 60-80% of instances of each class into the development set for "residential area", and 40-80% for "home". In the split process, the recordings were repeatedly and randomly assigned to the sets until the event-wise split conditions were met for all classes. Extra care was taken that the test subset did not contain classes unavailable in training.

Metrics

The metrics used for the ranking in the evaluation campaigns have to be well-established, already familiar from similar applications, and reproducible, ideally already having an available open-source implementation. In addition, the metric should be easy to understand for the participants, in order to make development and system optimization straightforward for them. DCASE 2016 and 2017 tasks used segment-based error rate calculated in one-second segments as primary ranking metric, and segment-based F-score calculated in same segments was used as secondary metric. The details of the metrics have been discussed in Section 4.3. In the evaluation, the folds in the cross-validation setup were treated as a single experiment and intermediate statistics of the metrics were accumulated over all folds instead of averaging fold-wise metric values [113]. This evaluation scheme gave equal weight to each individual sound instance in each segment minimizing the effect of folds having different numbers of event instances [49]. The two metrics used together provided a more comprehensive performance estimate for the systems and allowed more an in-depth analysis of their behavior. These metrics were proposed for SED performance evaluation in [113] along with an open-source Python toolbox[1] providing their implementation. The metrics have since become standard metrics in the research field for SED performance evaluation, partly due to their usage in the DCASE challenges.

Baseline Systems

The role of baseline systems in the evaluation campaign is two-fold. During the task preparation stage, they are used to verify the task setup and the performance generalization across development and evaluation datasets. During the challenge, the results of the baseline system for the development dataset gave participants a reference performance level for the task, and the actual open-source implementation allowed entry-level researchers to set up their development environment. The implementation also helped researchers from neighboring fields to start working on sound event detection by providing ready-made audio and metadata processing pipelines for the task.

The baseline system for DCASE 2016 was a polyphonic SED system based on the class-wise activity detection approach discussed earlier in Section 4.6.2. The system

[1] `https://github.com/TUT-ARG/sed_eval`

was implemented in Python[2] and Matlab environments[3] to serve as large as possible audience. The DCASE 2016 campaign showed a strong emergence of deep learning based approaches. To follow the trend, the baseline system of DCASE 2017 was based on feedforward neural network (a multilayer perceptron) capable of a multi-class multi-label classification per analysis frame [106] and implemented in Python[4]. The network output layer contained 6 sigmoid units that could be active at the same time and indicate activity of overlapping sound event classes. The detection was performed by integrating the classification decision for each class per frame using a sliding median window of 0.54 s length with a 20 ms hop. Log mel-band energies were extracted for the audio signal using 40 mel bands in 40 ms analysis frames with 50% overlap. To capture temporal information into the feature vector, five consecutive frames were concatenated within a context window, and the resulting 200-valued feature vector was fed into the network. The network consisted of two fully-connected layers of 50 hidden units each.

Analysis of Results

Teams participating to the challenge were allowed to submit a maximum of four separate systems. This allowed teams to test variations of one single system or to develop different approaches. The system outputs of each system for the evaluation dataset were submited to the task organizers, and these systems were described in a technical report that was also submitted as part of the challenge participation. In DCASE 2017, basic attributes of each system were also collected in a machine readable format to allow easier meta analysis of the submissions. The challenge results and all technical reports were published online on DCASE website[5].

The majority of the submitted systems outperformed the provided baseline systems, as expected due to their relative simplicity. In DCASE 2016 most of the submitted systems were based on deep learning; specially, the best systems (top-7) were based on DNN, RNN, or fusions of various deep learning architectures. Other approaches used in the submitted systems included random forest and HMM-based solutions. Mel-scale based acoustic features, MFCCs and log mel energies, were the most common choice. The best performing system was based on spatial and harmonic features extracted from binaural signals and a recurrent neural network using long short-term memory (LSTM) to incorporate contextual information from previous frames into detection [4]. The performance metrics for this system were 0.80 error rate and 47.8% F-score, outperforming the baseline system having error

[2] `https://github.com/TUT-ARG/DCASE2016-baseline-system-python`

[3] `https://github.com/TUT-ARG/DCASE2016-baseline-system-matlab`

[4] `https://github.com/TUT-ARG/DCASE2017-baseline-system`

[5] `http://dcase.community/`

rate of 0.88 and F-score of 34.3%.

In DCASE 2017 there was not much diversity in the system characteristics, as the majority of the systems used single-channel audio, mel-based representations as features, and deep learning based classifiers. The best performing system was based on CRNNs and reached error rate of 0.79 and 41.7% F-score, clearly outperforming the baseline system in error rate of 0.94 and having similar performance performance on F-score (42.8%). In order to have a statistical analysis of the SED results, [109] introduced the calculation of 95% confidence intervals for these metrics using a jackknife resampling procedure. This procedure gives a coarse approximation of confidence intervals without any knowledge of the underlying distribution of the parameters. Based on the confidence intervals, the four top-ranked systems did not have significantly different performance in terms of metrics. Interestingly, the majority of the systems had similar performance when measuring in F-score, around 41%, and the biggest differences could be seen in the error rate. The system outputs from DCASE 2016 [125] and 2017 [110] have been published to allow their re-evaluation in the future with new metrics.

4.8.2 Challenge Impact

The SED tasks gathered significant interest in DCASE 2016 and 2017 challenges. In 2016, there were 16 systems submitted from 12 participating teams, and in 2017 the number of submitted systems increased to 35 while the number of participating teams remained the same. After the challenges, some of the participants published extended studies based on their submitted systems [3, 156, 199]. The open datasets with cross-validation setups and open-source evaluation toolboxes released under the challenge allowed researchers to compare their work outside the challenge to all the submitted works from the challenge. This stabilized the research in the field by providing a uniform reference point across a large number of publications. Thereafter, other research groups started collecting and publishing open datasets, and this accelerated further the development of state-of-the-art methods. As a result, sound event detection as a research topic has grown exponentially in recent years. The impact of the various challenges on the number of yearly SED related publications is very strong, as can be seen in Figure 4.24. After DCASE 2016 challenge, the publication count has grown steadily from 371 to 1221. At the same time, DCASE SED task organization related publications [P6], [P7], [113], [109], and [106] have collected over 1000 total citations between 2016 and 2020 according to Google Scholar (retrieved January 2021).

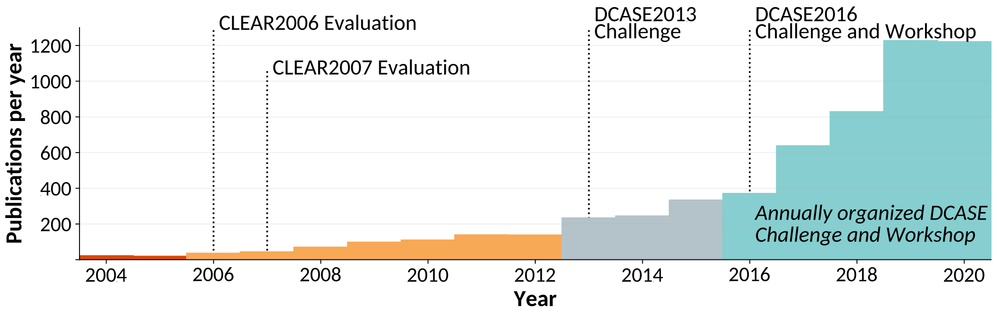


Figure 4.24 Sound event detection related publications per year according to Google Scholar (retrieved January 2021).

4.9 Discussion

The work presented in this chapter covers a variety of approaches at different stages of a sound event detection system, from model training strategies to system output design.

Acoustic Model Training

The studies presented in publications [P1]–[P4] and [P6] proposed methods for assigning audio segments for acoustic model training either directly from mixture signals or from audio streams acquired from the pre-processing stage using sound source separation. The main contribution of these methods was to enable the systems to be trained using recordings captured in multi-source acoustic environments, recorded in similar acoustic conditions as the ones in which the system would be evaluated and in the end deployed. This is important in the process of creating noise-robust detection systems that would provide good performance in real-life applications. These studies answer research question Q2 (see Section 1.2) about how to train acoustic models with material containing overlapping sounds.

The presented methods for using mixture signals for model training are still very topical. State-of-the-art SED approaches based on deep learning are usually directly assigning segments from mixture signals to classes when training models, as deep learning techniques are good at learning complex models with associations and relationships from the data given a sufficiently large dataset [24]. In addition, the recent development of deep learning-based sound source separation for arbitrary sound classes has created new sound source separation methods well suited for everyday environments [83, 182]. Hence recently the universal sound separation approaches have re-emerged as pre-processing techniques for SED to tackle interference of overlapping sounds in the mixture signals [179].

Sound Event Detection

The work in [P1] focused on the feasibility of the proposed HMM-based approach for sound event detection. The main contribution of this work was to show for the first time that sound event detection is possible with a large set of event classes in many types of everyday environments. This was also the first work to tackle the research question Q1 about how to implement an SED system for a large number of classes. The evaluation results were promising, but at the same time, they highlighted a clear need for polyphonic detection to match the situations encountered in complex everyday environments. Furthermore, analysis of results in context-dependent and context-independent settings using count-based event priors highlighted the benefits of the context-dependent detection approach. In the subsequent publications [P2]–[P4] and [P6], the proposed detection approach was extended to polyphonic detection.

The studies presented in publications [P2]–[P4] focused on polyphonic detection while maintaining focus on research question Q1 about the implementation of a SED system for a large number of classes. The main contribution of these publications is that they introduced pioneering work that opened up the development of sound event detection towards real-life applications capable of detecting multiple events at the same time. As a consequence, these publications steered the research in the field away from the constrained indoor acoustic scenes, such as "meeting room" often used before, into a wide range of everyday environments. The proposed polyphonic detection systems were implemented using traditional pattern classification techniques that are not designed for multi-label output. However, the proposed methods included extensions to these techniques which made them feasible for polyphonic detection by producing multiple outputs. Publications [P2] and [P3] also introduced new metrics for measuring the system performance in case of multiple outputs, which have since become standard metrics in this domain.

In recent years, deep learning has become the dominant approach for state-of-the-art SED systems due to their superior performance compared to the traditional pattern classification techniques [24, 87, 187]. Deep neural networks are capable of learning complex relationships in the data given sufficient learning material, and as result, they usually provide considerably more robust acoustic models. Deep neural networks can be easily used to implement the multi-label classifier needed for polyphonic detection by assigning individual neurons in the output layer to indicate activity of different event classes. As a consequence, there is no need for additional techniques to cover the gap between the monophonic classification capability and the polyphonic requirement of the detection as was the case with the traditional methods. Nevertheless, systems based on traditional approaches such as HMMs can still provide additional modeling capabilities in the form of temporal decoding [70], similar to the way they are used in deep learning-based speech recognition to represent language models. Combining deep neural networks with HMMs allows explicit modeling of knowledge about the data which is not included in the acoustic models, such as prior probabilities or probabilities of event sequences, which are currently not represented in the state-of-the-art SED systems.

Contextual Information

Part of the work presented in this thesis focused on exploiting contextual information in audio content analysis. Publications [P2], [P5] and [P6] included studies in which acoustic scene classification was approached as a standalone task, whereas publication [P2] focused on incorporating contextual information provided by the acoustic scene classification into sound event detection. The main contribution of [P2] was to show that contextual information together with prior information about the event distributions within scene classes increases the performance of sound event detection significantly. The proposed approach tackles research question Q1 by dissecting the problem into smaller and more easily manageable ones. The work in [P5] using event histograms to recognize acoustic scenes answers research question Q4 on how to differentiate environments having similar acoustic properties.

Current state-of-the-art content analysis systems that are based on deep learning are generally robust enough to be able to operate in a wide range of different context types as long as the material used to train the system is diverse enough. For sound event detection, CRNN-based systems use recurrent layers to capture long temporal structures in the data, essentially modeling contextual information [24]. For acoustic scene classification, neural network based systems do not explicitly model sound events in the scene, but layers of the network can be regarded to model latent attributes of the scene that are related to the prominent sound events in the audio [184].

Open Science

DCASE evaluation campaigns have had an important impact on the growth of the research field. DCASE challenge and workshop have helped to create a thriving research community with members from academia and industry. The author has acted as task coordinator for SED tasks in 2016 and 2017, and acoustic scene classification task from 2016 to 2020. In this role, he has published 26 open datasets to be used in these challenge tasks, released ten open-source baseline systems, authored the evaluation toolbox for SED, data management utility toolbox for DCASE [6], implemented the evaluation of the submissions, and maintained the DCASE community website. He

[6] https://github.com/DCASE-REPO/dcase_util

has also contributed to over ten publications related to these challenge tasks.

The work for this thesis was done during a pivotal period in the research field. Open science has increased the transparency and accessibility of research: open datasets, open-source toolboxes, and system implementations are nowadays standard in the field. The deep learning methods have increased the achievable performance substantially compared to the traditional classification methods as the available learning data amounts have increased. Furthermore, the deep learning methods have enabled researchers to tackle more complex content analysis problems such as weakly supervised learning [126, 203] or multi-task learning [3, 135, 177]. This has enabled sound event detection systems that can be trained with weakly labelled material [126], and systems that are able to localize and detect sound events at the same time [3]. One of the recent trends in the field is learning from low-resource data; one-shot and few-shot learning methods aim at learning robust models based on a limited set of examples [86, 196, 205], and zero-shot learning uses semantic information embedded into textual labels given to sounds to facilitate learning without audio examples [201].

5 CONCLUSIONS

This thesis presented computational methods for audio content analysis of everyday environments. The work on content analysis focused especially on sound events, as they have an important role in understanding what is happening in the environment. The core contributions of the work are related to sound event detection using a large set of sound event classes in real-life environments where multiple sound events occur simultaneously. The work presented in this thesis was the first to extend the sound event detection methods available at the time with approaches capable of producing polyphonic outputs, which in turn increased performance and applicability of the methods in everyday environments substantially.

Audio captured in everyday environments contains a wide range of sounds that are overlapping in time and frequency, and this poses a major challenge for computational content analysis systems. Part of the overlapping sounds can be always seen as interfering sounds that make the learning of acoustic models challenging. The presented work handled this interference in the pre-processing stage of the analysis system by using sound source separation techniques to decompose a mixture signal into individual sound sources. The inter-class acoustic variability of the sound events introduces an extra challenge for modeling, especially when the analysis system aims to recognize a large set of sound event classes, because one cannot make strong assumptions about the temporal or spectral structure of the sounds when selecting the approach. The work presented in this thesis used HMM-based acoustic models with a three-stage left-to-right topology, which were found to provide good modeling of the temporal structure for a large variety of sound event classes.

The main objective of the thesis was to develop methods for sound event detection with a large set of sound events and varying degree of polyphony, and to solve the handling of overlapping sounds. The solution started with an HMM-based approach for prominent sound event detection for a large set of classes, proposed in [P1], which was then extended into polyphonic detection. The polyphonic detection was implemented using multiple Viterbi iterations [P2] or by using sound source separation [P3], [P4]. These polyphonic SED systems were not based on multi-label classification, instead they used a collection of generative classifiers to produce multiple labels for the same time instance, but nevertheless achieved at least double the detection performance than the monophonic one. Later in [P7], the polyphonic detection problem was approached using class-wise activity detection; the activity of each event class was detected independently, and class-wise event sequences were merged to produce the polyphonic system output. The approach using sound source separation as pre-processing was found to produce the best performance, improving segment-based detection F-score from 15 to 45% when evaluated in one second segments. Furthermore, the ability to detect multiple simultaneous sound events brought the methods closer to the practical applicability of SED systems in real-life scenarios.

A second objective of the thesis was to develop an evaluation procedure for polyphonic sound event detection by defining appropriate metrics. New metrics for the evaluation of polyphonic sound event detection were proposed in [P2] and [P3] along with the development of the polyphonic sound event detection systems, taking into account the polyphony. The proposed segment-based metrics are more suitable for polyphonic SED than the previously used ones originating from the CLEAR evaluation campaign because the overlapping sound events are compared one by one between reference and system output in fixed time segments instead of being compared based on the combinations of reference and system output events, which is dependent on the particular system that is evaluated. In addition, the new metrics introduced rigorous definitions for the process of determining the correct and erroneous cases, creating a consistent procedure for evaluation. In recent years, the proposed metrics have become the standard metrics in the research field.

A third objective of the thesis was to study sound events as a constituent parts of the acoustic scene. To this end, the studies included in the thesis used contextual information provided by the co-occurrence of sound events. In [P2], the detection of similarly sounding sound events was approached by identifying first the acoustic scene they were occurring in and narrowing down the selection of possible sound event classes based on this information. This approach enabled the use of context-dependent acoustic models, trained with material recorded in similar conditions as the tested signal, and the use of context-dependent event priors. The context-dependent SED increased the detection performance especially when count-based priors were used. The absolute performance increase for monophonic SED with context-dependent approach was approximately 3% when evaluated in one second segments. The context-dependent approach was used in subsequent publications [P3] and [P4]. In [P5], acoustic scenes were identified based on the distinctive and scene-specific sound events occurring in the scene, and the proposed event histogram-based classification approach achieved comparable performance to the traditional approach which models the general acoustic characteristics of the scenes. Fusion of these two approaches showed a significant increase in performance.

Finally, the last objective of the thesis was to develop tools for open research in the field. This was undeniably achieved through the efforts made for the publication of open datasets and open-source tools. The growth of the research interest in this field was supported to a great extent by the DCASE evaluation campaigns, to which the work presented in this thesis has contributed directly. The work has standardized evaluation protocols and metrics [P7] for SED, and released open datasets and benchmark systems [P6].

The follow-up of the work presented in this thesis includes many studies tackling specific problems related to polyphonic sound event detection and environmental audio content analysis. Contributions related to sound event detection have focused on topics such as deep neural network architectures [22, 23, 24], sound events co-occurrence modeling [78, 112], coupled non-negative matrix factorization [108], and sound event envelope estimation [102]. Active learning approaches [207, 208, 209] have shown good results in decreasing by 80-90% the required amount of manual data annotation when training acoustic models for sound classification or sound event detection. In contrast to the main objective of this thesis, some follow-up work has dealt with applications focusing on achieving the highest possible performance for a very limited set of sound classes: rare sound event detection [106], environmental noise monitoring with prominent noise source recognition [100], and audio-based epileptic seizure detection [5]. Recently, the research related to SED has been extended to include spatial localization of the sound sources, for which new metrics that measure jointly localization and detection performance were proposed [107, 144]. In addition to SED related contributions, problems related to acoustic scene classification have been addressed as well: generalization to different recording devices [75, 123], low-complexity system design [75], open set classification [124], human and machine classification performance assessment [117], audio-based city classification [11], audio-visual scene classification [194].

Even though the research field is over a decade old, we have just started to identify and exploit the possibility for real applications that audio content analysis of everyday environments enables. The research directions emerging in the field are raised by specific needs or shortcomings related to the applications. The quality and amount of the learning data are currently in focus: how to learn robust models with low amount of annotated data, with labeled but noisy data, weakly labeled data, or with large amounts of unlabeled data. In addition, the research in the field is currently expanding towards multi-task approaches, such as SELD, as well as approaches taking account multi-modal information. The multi-modal approaches use, for example, semantic information from text used to describe audio, location information, or movement information such as acceleration. As the research results are getting more applicable to real-life applications, privacy-related matters are also increasingly studied.

REFERENCES

[1] J. Abeßer. A Review of Deep Learning Based Methods for Acoustic Scene Classification. *Applied Sciences* 10.6 (Mar. 2020), 2020.

[2] S. Adavanne, P. Pertilä and T. Virtanen. Sound Event Detection Using Spatial Features and Convolutional Recurrent Neural Network. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, 771–775.

[3] S. Adavanne, A. Politis, J. Nikunen and T. Virtanen. Sound Event Localization and Detection of Overlapping Sources Using Convolutional Recurrent Neural Networks. *IEEE Journal of Selected Topics in Signal Processing* 13.1 (2019), 34–48.

[4] S. Adavanne, G. Parascandolo, P. Pertila, T. Heittola and T. Virtanen. Sound Event Detection in Multichannel Audio Using Spatial and Harmonic Features. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*. Sept. 2016, 6–10.

[5] I. M. N. Ahsan, C. Kertesz, A. Mesaros, T. Heittola, A. Knight and T. Virtanen. Audio-Based Epileptic Seizure Detection. *27th European Signal Processing Conference (EUSIPCO)*. 2019, 1–5.

[6] P. K. Atrey, N. C. Maddage and M. S. Kankanhalli. Audio based event detection for multimedia surveillance. *2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Vol. 5. IEEE. 2006, V–V.

[7] J. A. Ballas. Common factors in the identification of an assortment of brief everyday sounds. *Journal of Experimental Psychology: Human Perception and Performance* 19.2 (1993), 250.

[8] J. A. Ballas and T. Mullins. Effects of context on the identification of everyday sounds. *Human Performance* 4.3 (1991), 199–219.

[9] D. Barchiesi, D. Giannoulis, D. Stowell and M. D. Plumbley. Acoustic Scene Classification: Classifying environments from the sounds they produce. *IEEE Signal Processing Magazine* 32.3 (2015), 16–34.

[10] D. Battaglino, L. Lepauloux, L. Pilati and N. Evans. Acoustic context recognition using local binary pattern codebooks. *2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. Oct. 2015, 1–5.

[11] H. L. Bear, T. Heittola, A. Mesaros, E. Benetos and T. Virtanen. City classification from multiple real-world sound scenes. English. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. Oct. 2019, 11–15.

[12] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz and H. Doraiswamy. SONYC: A System for Monitoring, Analyzing, and Mitigating Urban Noise Pollution. *Communications of the ACM* 62.2 (Feb. 2019), 68–77.

[13] J. P. Bello, C. Mydlarz and J. Salamon. Sound Analysis in Smart Cities. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 373–397.

[14] C. M. Bishop. *Pattern Recognition and Machine Learning*. New York, USA: Springer, 2006.

[15] J. K. Bizley and Y. E. Cohen. The what, where and how of auditory-object perception. *Nature Reviews. Neuroscience* 14.10 (Oct. 2013), 693–707.

[16] A. S. Bregman. *Auditory Scene Analysis*. Cambridge, MA, USA: The MIT Press, 1990.

[17] A. Brown, J. Kang and T. Gjestland. Towards standardization in soundscape preference assessment. *Applied Acoustics* 72.6 (2011), 387–392.

[18] J. C. Brown. Calculation of a constant Q spectral transform. *The Journal of the Acoustical Society of America* 89.1 (1991), 425–434.

[19] T. Butko, C. Canton-Ferrer, S. C., G. X., N. C., H. J. and C. J. Two-source Acoustic Event Detection and Localization: Online Implementation in a Smart-Room. *19th European Signal Processing Conference (EUSIPCO)*. 2011, 1317–1321.

[20] T. Butko, C. Canton-Ferrer, C. Segura, X. Giró, C. Nadeu, J. Hernando and J. R. Casas. Acoustic Event Detection Based on Feature-Level Fusion of Audio and Video Modalities. *EURASIP Journal on Advances in Signal Processing* 2011.1 (Feb. 2011).

[21] E. Çakir and T. Virtanen. End-to-End Polyphonic Sound Event Detection Using Convolutional Recurrent Neural Networks with Learned Time-Frequency Representation Input. *2018 International Joint Conference on Neural Networks (IJCNN)*. 2018, 1–7.

[22] E. Çakır, T. Heittola, H. Huttunen and T. Virtanen. Multi-Label vs. Combined Single-Label Sound Event Detection With Deep Neural Networks. *23rd European Signal Processing Conference (EUSIPCO)*. 2015.

[23] E. Çakır, T. Heittola, H. Huttunen and T. Virtanen. Polyphonic Sound Event Detection Using Multi Label Deep Neural Networks. *The International Joint Conference on Neural Networks 2015 (IJCNN 2015)*. 2015.

[24] E. Çakır, G. Parascandolo, T. Heittola, H. Huttunen and T. Virtanen. Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection. *Transactions on Audio, Speech and Language Processing* 25.6 (June 2017), 1291–1303.

[25] B. T. Carroll, D. V. Anderson, W. Daley, S. Harbert, D. F. Britton and M. W. Jackwood. Detecting symptoms of diseases in poultry through audio signal processing. *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. 2014, 1132–1135.

[26] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes and M. Slaney. Content-Based Music Information Retrieval: Current Directions and Future Challenges. *Proceedings of the IEEE* 96.4 (Apr. 2008), 668–696.

[27] M. Casey. General sound classification and similarity in MPEG-7. *Organised Sound* 6.2 (2001), 153–164.

[28] P. Chahuara, A. Fleury, F. Portet and M. Vacher. On-line human activity recognition from audio and home automation sensors: Comparison of sequential and non-sequential models in realistic Smart Homes 1. *Journal of Ambient Intelligence and Smart Environments* 8.4 (2016), 399–422.

[29] J. Chen, A. H. Kam, J. Zhang, N. Liu and L. Shue. Bathroom activity monitoring based on sound. *International Conference on Pervasive Computing*. Springer. 2005, 47–61.

[30] C. Clavel, T. Ehrette and G. Richard. Events Detection for an Audio-Based Surveillance System. *IEEE International Conference on Multimedia and Expo*. 2005, 1306–1309.

[31] M. Cooke and D. Ellis. The auditory organization of speech and other sources in listeners and computational models. *Speech Communication* 35.3-4 (2001), 141–177.

[32] J. Cramer, H. Wu, J. Salamon and J. P. Bello. Look, Listen, and Learn More: Design Choices for Deep Audio Embeddings. *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2019, 3852–3856.

[33] M. Crocco, M. Cristani, A. Trucco and V. Murino. Audio Surveillance: A Systematic Review. *ACM Computing Surveys* 48.4 (Feb. 2016).

[34] S. B. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28 (4 1980), 357–366.

[35] A. P. Dempster, N. M. Laird and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B* 39.1 (1977), 1–38.

[36] J. Dennis, H. D. Tran and E. S. Chng. Image Feature Representation of the Subband Power Distribution for Robust Sound Event Classification. *IEEE Transactions on Audio, Speech, and Language Processing* 21.2 (Feb. 2013), 367–377.

[37] A. Diment, T. Heittola and T. Virtanen. Semi-supervised Learning for Musical Instrument Recognition. *21st European Signal Processing Conference (EUSIPCO).* Sept. 2013.

[38] H. M. Do, W. Sheng, M. Liu and Senlin Zhang. Context-aware sound event recognition for home service robots. *2016 IEEE International Conference on Automation Science and Engineering (CASE).* 2016, 739–744.

[39] K.-L. Du and M. N. Swamy. *Neural Networks and Statistical Learning.* London, UK: Springer-Verlag London, 2013.

[40] X. Du, L. Carpentier, G. Teng, M. Liu, C. Wang and T. Norton. Assessment of Laying Hens' Thermal Comfort Using Sound Technology. *Sensors* 20.2 (Jan. 2020), 473.

[41] D. Dubois. Categories as acts of meaning: The case of categories in olfaction and audition. *Cognitive Science Quaterly* 1.1 (2000), 35–68.

[42] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* New York, NY, USA: John Wiley & Sons, 1973.

[43] D. Ellis. Prediction-driven computational auditory scene analysis. PhD thesis. MIT Media Laboratory, Cambridge, Massachusetts, 1996.

[44] A. J. Eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho and J. Huopaniemi. Audio-based context recognition. *IEEE Transactions on Audio, Speech, and Language Processing* 14.1 (Jan. 2006), 321–329.

[45] M. A. T. Figueiredo, J. M. N. Leitão and A. K. Jain. On Fitting Mixture Models. *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Ed. by E. R. Hancock and M. Pelillo. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, 54–69.

[46] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio and M. Vento. Audio Surveillance of Roads: A System for Detecting Anomalous Sounds. *IEEE Transactions on Intelligent Transportation Systems* 17.1 (2016), 279–288.

[47] E. Fonseca, M. Plakal, D. P. Ellis, F. Font, X. Favory and X. Serra. Learning sound event classifiers from web audio with noisy labels. *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, 21–25.

[48] E. Fonseca, M. Plakal, F. Font, D. P. Ellis, X. Favory, J. Pons and X. Serra. General-purpose tagging of Freesound audio with AudioSet labels: task description, dataset, and baseline. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*. Nov. 2018, 69–73.

[49] G. Forman and M. Scholz. Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *ACM SIGKDD Explorations Newsletter* 12.1 (2010), 49–57.

[50] P. Foster, S. Sigtia, S. Krstulovic, J. Barker and M. D. Plumbley. CHiME-Home: A Dataset for Sound Source Recognition in a Domestic Environment. *2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2015, 1–5.

[51] S. Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 29.2 (1981), 254–272.

[52] M. Gales and S. Young. The Application of Hidden Markov Models in Speech Recognition. *Foundations and Trends in Signal Processing* 1.3 (Jan. 2008), 195–304.

[53] A. Gasc, J. Sueur, F. Jiguet, V. Devictor, P. Grandcolas, C. Burrow, M. Depraetere and S. Pavoine. Assessing biodiversity with sound: Do acoustic diversity indices reflect phylogenetic and functional diversities of bird communities?: *Ecological Indicators* 25 (2013), 279–287.

[54] W. W. Gaver. Auditory Icons: Using Sound in Computer Interfaces. *Human-Computer Interaction* 2.2 (June 1986), 167–177.

[55] W. W. Gaver. How Do We Hear in the World? Explorations in Ecological Acoustics. *Ecological Psychology* 5.4 (1993), 285–313.

[56] W. W. Gaver. What in the World Do We Hear? an Ecological Approach to Auditory Event Perception. *Ecological Psychology* 5.1 (1993), 1–29.

[57] J. T. Geiger, B. Schuller and G. Rigoll. Large-scale audio feature extraction and SVM for acoustic scene classification. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. Oct. 2013, 1–4.

[58] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal and M. Ritter. Audio set: An ontology and human-labeled dataset for audio events. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, 776–780.

[59] O. Gencoglu, T. Virtanen and H. Huttunen. Recognition of acoustic events using deep neural networks. *22nd European Signal Processing Conference (EUSIPCO)*. Sept. 2014, 506–510.

[60] X. Glorot, A. Bordes and Y. Bengio. Deep Sparse Rectifier Neural Networks. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by G. Gordon, D. Dunson and M. Dudík. Vol. 15. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, Apr. 2011, 315–323.

[61] S. Goetze, J. Schroder, S. Gerlach, D. Hollosi, J.-E. Appell and F. Wallhoff. Acoustic monitoring and localization for social care. *Journal of Computing Science and Engineering* 6.1 (2012), 40–50.

[62] I. Goodfellow, Y. Bengio and A. Courville. *Deep Learning*. Cambridge, MA, USA: The MIT Press, 2016.

[63] M. Grassi, M. Pastore and G. Lemaitre. Looking at the world with your ears: How do we get the size of an object from its sound?: *Acta Psychologica* 143.1 (2013), 96–104.

[64] C. Guastavino. Categorization of environmental sounds. *Canadian Journal of Experimental Psychology* 61.1 (2007), 54–63.

[65] C. Guastavino. Everyday Sound Categorization. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 183–213.

[66] B. Gygi. Factors in the Identification of Environmental Sounds. PhD thesis. Indiana University, July 2001.

[67] B. Gygi, G. R. Kidd and C. S. Watson. Spectral-temporal factors in the identification of environmental sounds. *The Journal of the Acoustical Society of America* 115.3 (2004), 1252–1265.

[68] B. Gygi, G. R. Kidd and C. S. Watson. Similarity and categorization of environmental sounds. *Perception & Psychophysics* 69.6 (2007), 839–855.

[69] B. Gygi and V. Shafiro. Environmental sound research as it stands today. *Proceedings of Meetings on Acoustics* 1.1 (2007).

[70] T. Hayashi, S. Watanabe, T. Toda, T. Hori, J. Le Roux and K. Takeda. BLSTM-HMM hybrid system combined with sound activity detection network for polyphonic Sound Event Detection. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, 766–770.

[71] T. Heittola, A. Klapuri and T. Virtanen. Musical Instrument Recognition in Polyphonic Audio Using Source-Filter Model for Sound Separation. *Proceedings 10th International Society for Music Information Retrieval Conference (ISMIR)*. 2009, 327–332.

[72] T. Heittola, E. Çakır and T. Virtanen. The Machine Learning Approach for Analysis of Sound Scenes and Events. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 13–40.

[73] T. Heittola and A. Klapuri. TUT Acoustic Event Detection System 2007. *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. Ed. by R. Stiefelhagen, R. Bowers and J. Fiscus. Cham, Switzerland: Springer Verlag, 2008, 364–370.

[74] T. Heittola, A. Mesaros, D. Korpi, A. Eronen and T. Virtanen. Method for Creating Location-Specific Audio Textures. *EURASIP Journal on Audio, Speech and Music Processing* 1.9 (2014).

[75] T. Heittola, A. Mesaros and T. Virtanen. Acoustic Scene Classification in DCASE 2020 Challenge: Generalization Across Devices and Low Complexity Solutions. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*. Nov. 2020, 56–60.

[76] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss and K. Wilson. CNN architectures for large-scale audio classification. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Mar. 2017, 131–135.

[77] O. Houix, G. Lemaitre, N. Misdariis, P. Susini and I. Urdapilleta. A lexical analysis of environmental sound categories. *Journal of Experimental Psychology: Applied* 18.1 (2012), 52.

[78] G. Huang, T. Heittola and T. Virtanen. Using Sequential Information in Polyphonic Sound Event Detection. *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC).* Sept. 2018, 291–295.

[79] G. W. Humphreys, M. J. Riddoch and C. J. Price. Top-Down Processes in Object Identification: Evidence from Experimental Psychology, Neuropsychology and Functional Anatomy. *Philosophical Transactions: Biological Sciences* 352.1358 (1997), 1275–1282.

[80] A. Jansen, J. F. Gemmeke, D. P. W. Ellis, X. Liu, W. Lawrence and D. Freedman. Large-scale audio event discovery in one million YouTube videos. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2017, 786–790.

[81] M. Janvier, X. Alameda-Pineda, L. Girinz and R. Horaud. Sound-event recognition with a companion humanoid. *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012).* 2012, 104–111.

[82] S. Jung, J. Park and S. Lee. Polyphonic Sound Event Detection Using Convolutional Bidirectional LSTM and Synthetic Data-based Transfer Learning. *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2019, 885–889.

[83] I. Kavalerov, S. Wisdom, H. Erdogan, B. Patton, K. Wilson, J. Le Roux and J. R. Hershey. Universal Sound Separation. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA).* Oct. 2019, 175–179.

[84] T. Kim, J. Lee and J. Nam. Comparison and Analysis of SampleCNN Architectures for Audio Classification. *IEEE Journal of Selected Topics in Signal Processing* 13.2 (May 2019), 285–297.

[85] T. Kobayashi and J. Ye. Acoustic feature extraction by statistics based local binary pattern for environmental sound classification. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* May 2014, 3052–3056.

[86] Y. Koizumi, M. Yasuda, S. Murata, S. Saito, H. Uematsu and N. Harada. SPIDERnet: Attention Network For One-Shot Anomaly Detection In Sounds. *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2020, 281–285.

[87] Q. Kong, Y. Xu, I. Sobieraj, W. Wang and M. D. Plumbley. Sound event detection and time–frequency segmentation from weakly labelled data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.4 (2019), 777–787.

[88] F. Kraft, R. Malkin, T. Schaaf and A. Waibel. Temporal ICA Classification of Acoustic Events in a Kitchen Enviroment. *Proceedings of ICSLP-Interspeech*. 2005, 2689–2692.

[89] S. Krstulović. Audio Event Recognition in the Smart Home. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 335–371.

[90] A. Kumar, R. M. Hegde, R. Singh and B. Raj. Event detection in short duration audio using Gaussian Mixture Model and Random Forest Classifier. *21st European Signal Processing Conference (EUSIPCO)*. Sept. 2013, 1–5.

[91] A. Kumar and B. Raj. Audio Event Detection Using Weakly Labeled Data. *Proceedings of the 24th ACM International Conference on Multimedia*. 2016, 1038–1047.

[92] A. J. Kunkler-Peck and M. Turvey. Hearing shape. *Journal of Experimental Psychology: Human perception and Performance* 26.1 (2000), 279.

[93] G. Lafay, E. Benetos and M. Lagrange. Sound event detection in synthetic audio: Analysis of the DCASE 2016 task results. *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. Oct. 2017, 11–15.

[94] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86.11 (1998), 2278–2324.

[95] J. Lee, B. Noh, S. Jang, D. Park, Y. Chung and H.-H. Chang. Stress detection and classification of laying hens by sound analysis. *Asian-Australasian Journal of Animal Sciences* 28.4 (2015), 592.

[96] G. Lemaitre, N. Grimault and C. Suied. Acoustics and Psychoacoustics of Sound Scenes and Events. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 41–67.

[97] G. Lemaitre and L. M. Heller. Auditory perception of material is fragile while action is strikingly robust. *The Journal of the Acoustical Society of America* 131.2 (2012), 1337–1348.

[98] G. Lemaitre, O. Houix, N. Misdariis and P. Susini. Listener expertise and sound identification influence the categorization of environmental sounds. *Journal of Experimental Psychology: Applied* 16.1 (2010), 16.

[99] J. Li, L. Deng, R. Haeb-Umbach and Y. Gong. *Robust Automatic Speech Recognition*. Waltham, MA, USA: Academic Press, 2016.

[100] P. Maijala, S. Zhao, T. Heittola and T. Virtanen. Environmental Noise Monitoring Using Source Classification in Sensors. *Applied Acoustics* 129.6 (Jan. 2018), 258–267.

[101] M. Mandel, J. Salamon and D. P. W. Ellis. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019).* New York, NY, USA: New York University, Oct. 2019.

[102] I. Martin-Morato, A. Mesaros, T. Heittola, T. Virtanen, M. Cobos and F. J. Ferri. Sound event envelope estimation in polyphonic mixtures. *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* May 2019, 935–939.

[103] S. McAdams. The psychomechanics of real and simulated sound sources. *The Journal of the Acoustical Society of America* 107.5 (2000), 2792–2792.

[104] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. 2nd ed. Vol. 382. Hoboken, NJ, USA: John Wiley & Sons, 2008.

[105] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen and M. D. Plumbley. Detection and Classification of Acoustic Scenes and Events: Outcome of the DCASE 2016 Challenge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.2 (Feb. 2018), 379–393.

[106] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj and T. Virtanen. DCASE 2017 Challenge Setup: Tasks, Datasets and Baseline System. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017).* Nov. 2017, 85–92.

[107] A. Mesaros, S. Adavanne, A. Politis, T. Heittola and T. Virtanen. Joint Measurement of Localization and Detection of Sound Events. English. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA).* Oct. 2019, 328–332.

[108] A. Mesaros, O. Dikmen, T. Heittola and T. Virtanen. Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2015, 151–155.

[109] A. Mesaros, A. Diment, B. Elizalde, T. Heittola, E. Vincent, B. Raj and T. Virtanen. Sound event detection in the DCASE 2017 Challenge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.6 (June 2019), 992–1006.

[110] A. Mesaros, A. Diment, T. Heittola, B. Elizalde, E. Vincent, B. Raj and T. Virtanen. *DCASE2017 Challenge Submissions Package*. Zenodo, 2019. DOI: `10.5281/zenodo.2598364`.

[111] A. Mesaros, T. Heittola and D. Ellis. Datasets and Evaluation. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 147–179.

[112] A. Mesaros, T. Heittola and A. Klapuri. Latent Semantic Analysis in Sound Event Detection. *19th European Signal Processing Conference (EUSIPCO)*. 2011, 1307–1311.

[113] A. Mesaros, T. Heittola and T. Virtanen. Metrics for Polyphonic Sound Event Detection. *Applied Sciences* 6.6 (2016).

[114] A. Mesaros, T. Heittola and T. Virtanen. *TUT Acoustic scenes 2016, Development dataset*. Zenodo, 2016. DOI: `10.5281/zenodo.45739`.

[115] A. Mesaros, T. Heittola and T. Virtanen. *TUT Acoustic scenes 2016, Evaluation dataset*. Zenodo, 2016. DOI: `10.5281/zenodo.165995`.

[116] A. Mesaros, T. Heittola and T. Virtanen. *TUT Sound events 2016, Development dataset*. Zenodo, Feb. 2016. DOI: `10.5281/zenodo.45759`.

[117] A. Mesaros, T. Heittola and T. Virtanen. Assessment of Human and Machine Performance in Acoustic Scene Classification: DCASE 2016 Case Study. *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE Computer Society, 2017, 319–323.

[118] A. Mesaros, T. Heittola and T. Virtanen. *TUT Acoustic scenes 2017, Development dataset*. Zenodo, 2017. DOI: `10.5281/zenodo.400515`.

[119] A. Mesaros, T. Heittola and T. Virtanen. *TUT Acoustic scenes 2017, Evaluation dataset*. Zenodo, 2017. DOI: `10.5281/zenodo.1040168`.

[120] A. Mesaros, T. Heittola and T. Virtanen. *TUT Sound events 2016, Evaluation dataset*. Zenodo, Sept. 2017. DOI: `10.5281/zenodo.996424`.

[121] A. Mesaros, T. Heittola and T. Virtanen. *TUT Sound events 2017, Development dataset*. Zenodo, Mar. 2017. DOI: `10.5281/zenodo.814831`.

[122] A. Mesaros, T. Heittola and T. Virtanen. *TUT Sound events 2017, Evaluation dataset*. Zenodo, Nov. 2017. DOI: `10.5281/zenodo.1040179`.

[123] A. Mesaros, T. Heittola and T. Virtanen. A multi-device dataset for urban acoustic scene classification. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*. Nov. 2018, 9–13.

[124] A. Mesaros, T. Heittola and T. Virtanen. Acoustic Scene Classification in DCASE 2019 challenge: closed and open set classification and data mismatch setups. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*. Oct. 2019, 164–168.

[125] A. Mesaros, T. Heittola, T. Virtanen, E. Benetos, M. Lagrange, G. Lafay, P. Foster and M. D. Plumbley. *DCASE2016 Challenge Submissions Package*. Zenodo, 2017. DOI: `10.5281/zenodo.926660`.

[126] K. Miyazaki, T. Komatsu, T. Hayashi, S. Watanabe, T. Toda and K. Takeda. Weakly-Supervised Sound Event Detection with Self-Attention. *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, 66–70.

[127] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: The MIT Press, 2012.

[128] C. Mydlarz, J. Salamon and J. Bello. The implementation of low-cost urban acoustic monitoring devices. English (US). *Applied Acoustics* 117 (Feb. 2017), 207–218.

[129] A. Y. Ng and M. I. Jordan. On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes. *Advances in Neural Information Processing Systems 14*. Ed. by T. G. Dietterich, S. Becker and Z. Ghahramani. MIT Press, 2002, 841–848.

[130] M. E. Niessen, L. van Maanen and T. C. Andringa. Disambiguating Sounds through Context. *2008 IEEE International Conference on Semantic Computing*. 2008, 88–95.

[131] S. Ntalampiras and I. Potamitis. Transfer Learning for Improved Audio-Based Human Activity Recognition. *Biosensors* 8.3 (June 2018), 60.

[132] D. O'Shaughnessy. *Speech Communication: Human and Machine*. 2nd ed. New York, NY, USA: IEEE Press, 2000.

[133] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in Cognitive Sciences* 11.12 (2007), 520–527.

[134] A. V. Oppenheim, R. W. Schafer and J. R. Buck. *Discrete-Time Signal Processing*. Upper Saddle River, NJ, USA: Prentice Hall, 1999.

[135] A. Pankajakshan, H. Bear and E. Benetos. Onsets, Activity, and Events: A Multi-task Approach for Polyphonic Sound Event Modelling. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*. Oct. 2019, 174–178.

[136] G. Parascandolo, H. Huttunen and T. Virtanen. Recurrent neural networks for polyphonic sound event detection in real life recordings. *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE. 2016, 6440–6444.

[137] K. Pearson. Contributions to the Mathematical Theory of Evolution. *Philosophical Transactions of the Royal Society of London. A* 185 (1894), 71–110.

[138] V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi and T. Sorsa. Computational auditory scene recognition. *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 2. 2002, II-1941-II–1944.

[139] Y.-T. Peng, C.-Y. Lin, M.-T. Sun and K.-C. Tsai. Healthcare audio event classification using Hidden Markov Models and Hierarchical Hidden Markov Models. *IEEE International Conference on Multimedia and Expo*. June 2009, 1218–1221.

[140] J. Pickles. *An Introduction to the Physiology of Hearing*. 4th ed. Bingley, UK: Emerald, 2012.

[141] K. J. Piczak. Environmental sound classification with convolutional neural networks. *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP).* 2015, 1–6.

[142] M. D. Plumbley, C. Kroos, J. P. Bello, G. Richard, D. P. Ellis and A. Mesaros. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018).* Tampere, Finland: Tampere University of Technology. Laboratory of Signal Processing, 2018.

[143] G. E. Poliner and D. P. Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing* 2007 (2006), 1–9.

[144] A. Politis, A. Mesaros, S. Adavanne, T. Heittola and T. Virtanen. Overview and Evaluation of Sound Event Localization and Detection in DCASE 2019. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 684–698.

[145] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. New Jersey, USA: PTR Prentice-Hall Inc., 1993.

[146] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77.2 (Feb. 1989), 257–286.

[147] A. Rakotomamonjy and G. Gasso. Histogram of Gradients of Time–Frequency Representations for Audio Scene Classification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.1 (Jan. 2015), 142–153.

[148] D. Reynolds and R. Rose. Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Transactions on Speech and Audio Processing* 3.1 (1995), 72–83.

[149] S. Robertson. Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation* 60.5 (2004), 503–520.

[150] G. Roma, W. Nogueira and P. Herrera. Recurrence quantification analysis features for environmental sound recognition. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2013, 1–4.

[151] E. Rosch. Principles of Categorization. *Cognition and Categorization*. Ed. by E. Rosch and B. B. Lloyd. Hillsdale, NJ: Erlbaum, 1978, 27–48.

[152] M. Ryynänen and A. Klapuri. Polyphonic music transcription using note event modeling. *2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2005, 319–322.

[153] J. Salamon and J. P. Bello. Unsupervised feature learning for urban sound classification. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, 171–175.

[154] J. Salamon and J. P. Bello. Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification. *IEEE Signal Processing Letters* 24.3 (2017), 279–283.

[155] J. Salamon, C. Jacoby and J. P. Bello. A dataset and taxonomy for urban sound research. *Proceedings of the 22nd ACM International Conference on Multimedia*. 2014, 1041–1044.

[156] J. Schröder, N. Moritz, J. Anemüller, S. Goetze and B. Kollmeier. Classifier Architectures for Acoustic Scenes and Events: Implications for DNNs, TDNNs, and Perceptual Features from DCASE 2016. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.6 (2017), 1304–1314.

[157] J. Schröder, N. Moritz, M. R. Schädler, B. Cauchi, K. Adiloglu, J. Anemüller, S. Doclo, B. Kollmeier and S. Goetze. On the use of spectro-temporal features for the IEEE AASP challenge 'detection and classification of acoustic scenes and events'. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. Oct. 2013, 1–4.

[158] R. V. Shannon, F.-G. Zeng, V. Kamath, J. Wygonski and M. Ekelid. Speech recognition with primarily temporal cues. *Science* 270.5234 (1995), 303–304.

[159] M. Slaney. *Auditory toolbox*. Tech. rep. Technical Report 1998-010. Interval Research Corporation, 1998.

[160] E. E. Smith and D. L. Medin. *Categories and concepts*. Vol. 9. Cambridge, MA, USA: Harvard University Press, 1981.

[161] S. S. Stevens, J. Volkmann and E. B. Newman. A Scale for the Measurement of the Psychological Magnitude Pitch. *The Journal of the Acoustical Society of America* 8.3 (1937), 185–190.

[162] R. Stiefelhagen, H. K. Ekenel, C. Fugen, P. Gieselmann, H. Holzapfel, F. Kraft, K. Nickel, M. Voit and A. Waibel. Enabling Multimodal Human–Robot Interaction for the Karlsruhe Humanoid Robot. *IEEE Transactions on Robotics* 23.5 (2007), 840–851.

[163] R. Stiefelhagen, K. Bernardin, R. Bowers, J. Garofolo, D. Mostefa and P. Soundararajan. The CLEAR 2006 Evaluation. *Proceedings of the 1st International Evaluation Conference on Classification of Events, Activities and Relationships*. CLEAR'06. Southampton, UK: Springer-Verlag, 2006, 1–44.

[164] R. Stiefelhagen, R. Bowers and J. Fiscus. *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. Vol. 4625. Cham, Switzerland: Springer Verlag, 2008.

[165] D. Stowell, E. Benetos and L. F. Gill. On-Bird Sound Recordings: Automatic Acoustic Recognition of Activities and Contexts. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.6 (2017), 1193–1206.

[166] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange and M. Plumbley. Detection and Classification of Acoustic Scenes and Events. *Multimedia, IEEE Transactions on* 17.10 (Oct. 2015), 1733–1746.

[167] D. Stowell. Computational Bioacoustic Scene Analysis. *Computational Analysis of Sound Scenes and Events*. Ed. by T. Virtanen, M. D. Plumbley and D. Ellis. Cham, Switzerland: Springer Verlag, 2018, 303–333.

[168] B. T. Szabó, S. L. Denham and I. Winkler. Computational Models of Auditory Scene Analysis: A Review. *Frontiers in Neuroscience* 10 (2016), 524.

[169] N. Takahashi, M. Gygli and L. Van Gool. AENet: Learning Deep Audio Features for Video Analysis. *IEEE Transactions on Multimedia* 20.3 (2018), 513–524.

[170] A. Temko and C. Nadeu. Acoustic event detection in a meeting-room environment. *Pattern Recognition Letters* 30.14 (2009), 1281–1288.

[171] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu and M. Omologo. CLEAR evaluation of acoustic event detection and classification systems. *International Evaluation Workshop on Classification of Events, Activities and Relationships*. Springer. 2006, 311–322.

[172] A. Temko and C. Nadeu. Classification of acoustic events using SVM-based clustering schemes. *Pattern Recognition* 39.4 (2006), 682–694.

[173] A. Temko, C. Nadeu and J.-I. Biel. Acoustic event detection: SVM-based system and evaluation setup in CLEAR'07. *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. Ed. by R. Stiefelhagen, R. Bowers and J. Fiscus. Cham, Switzerland: Springer Verlag, 2007, 354–363.

[174] A. Temko, C. Nadeu, D. Macho, R. Malkin, C. Zieger and M. Omologo. Acoustic Event Detection and Classification. *Computers in the Human Interaction Loop*. Ed. by A. H. Waibel and R. Stiefelhagen. Springer London, 2009, 61–73.

[175] Term-weighting approaches in automatic text retrieval. *Information Processing Management* 24.5 (1988), 513–523.

[176] Y. Tokozume and T. Harada. Learning environmental sounds with end-to-end convolutional neural network. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, 2721–2725.

[177] N. Tonami, K. Imoto, M. Niitsuma, R. Yamanishi and Y. Yamashita. Joint Analysis of Acoustic Events and Scenes Based on Multitask Learning. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2019, 338–342.

[178] H. D. Tran and H. Li. Probabilistic distance SVM with Hellinger-Exponential Kernel for sound event classification. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2011, 2272–2275.

[179] N. Turpault, S. Wisdom, H. Erdogan, J. R. Hershey, R. Serizel, E. Fonseca, P. Seetharaman and J. Salamon. Improving Sound Event Detection in Domestic Environments using Sound Separation. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*. Nov. 2020, 205–209.

[180] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing* 10.5 (July 2002), 293–302.

[181] G. Tzanetakis, G. Essl and P. R. Cook. Audio Analysis using the Discrete Wavelet Transform. *Proceedings of the WSES International Conference Acoustics and Music: Theory and Applications (AMTA 2001)*. 2001, 318–323.

[182] E. Tzinis, S. Wisdom, J. R. Hershey, A. Jansen and D. P. W. Ellis. Improving Universal Sound Separation Using Sound Classification. *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* May 2020.

[183] M. Vacher, D. Istrate, L. Besacier, J.-F. Serignat and E. Castelli. Sound Detection and Classification for Medical Telesurvey. *2nd Conference on Biomedical Engineering*. Ed. by C. ACTA Press. Feb. 2004, 395–398.

[184] M. Valenti, S. Squartini, A. Diment, G. Parascandolo and T. Virtanen. A convolutional neural network approach for acoustic scene classification. *2017 International Joint Conference on Neural Networks (IJCNN).* 2017, 1547–1554.

[185] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci and A. Sarti. Scream and Gunshot Detection and Localization for Audio-Surveillance Systems. *Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance*. 2007, 21–26.

[186] N. J. Vanderveer. Ecological Acoustics: Human Perception of Environmental Sounds. PhD thesis. Cornell University, 1979.

[187] F. Vesperini, L. Gabrielli, E. Principi and S. Squartini. Polyphonic Sound Event Detection by Using Capsule Neural Networks. *IEEE Journal of Selected Topics in Signal Processing* 13.2 (2019), 310–322.

[188] T. Virtanen. Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria. *IEEE Transactions on Audio, Speech, and Language Processing* 15.3 (Mar. 2007), 1066–1074.

[189] T. Virtanen, A. Mesaros, T. Heittola, A. Diment, E. Vincent, E. Benetos and B. M. Elizalde. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017).* Tampere, Finland: Tampere University of Technology. Laboratory of Signal Processing, 2017.

[190] T. Virtanen, A. Mesaros, T. Heittola, M. Plumbley, P. Foster, E. Benetos and M. Lagrange. *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016).* Tampere, Finland: Tampere University of Technology. Department of Signal Processing, 2016.

[191] T. Virtanen, M. D. Plumbley and D. Ellis. *Computational Analysis of Sound Scenes and Events*. Cham, Switzerland: Springer Verlag, 2018.

[192] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* 13.2 (Apr. 1967), 260–269.

[193] D. Wang and G. J. Brown. *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. New York, NY, USA: Wiley-IEEE Press, 2006.

[194] S. Wang, A. Mesaros, T. Heittola and T. Virtanen. A Curated Dataset of Urban Scenes for Audio-Visual Scene Analysis. *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. accepted. IEEE. 2021.

[195] W. Wang. *Machine Audition: Principles, Algorithms and Systems*. Hershey, PA, USA: IGI Global, 2010.

[196] Y. Wang, J. Salamon, N. J. Bryan and J. Pablo Bello. Few-Shot Sound Event Detection. *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, 81–85.

[197] R. M. Warren. Auditory Perception and Speech Evolution. *Annals of the New York Academy of Sciences* 280.1 (1976), 708–717.

[198] F. Weninger and B. Schuller. Audio recognition in the wild: Static and dynamic classification on a real-world database of animal vocalizations. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. May 2011, 337–340.

[199] X. Xia, R. Togneri, F. Sohel and D. Huang. Confidence Based Acoustic Event Detection. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, 306–310.

[200] Xiaodan Zhuang, J. Huang, G. Potamianos and M. Hasegawa-Johnson. Acoustic fall detection using Gaussian mixture models and GMM supervectors. *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2009, 69–72.

[201] H. Xie and T. Virtanen. Zero-Shot Audio Classification Based On Class Label Embeddings. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2019, 264–267.

[202] M. Xu, C. Xu, L. Duan, J. S. Jin and S. Luo. Audio Keywords Generation for Sports Video Analysis. *ACM Transactions on Multimedia Computing, Communications, and Applications* 4.2 (2008), 1–23.

[203] Y. Xu, Q. Kong, W. Wang and M. D. Plumbley. Large-Scale Weakly Supervised Audio Classification Using Gated Convolutional Neural Network. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, 121–125.

[204] S. J. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev and P. Woodland. *The HTK Book Version 3.4*. Cambridge, UK: Cambridge University Press, 2006.

[205] S. Zhang, Y. Qin, K. Sun and Y. Lin. Few-Shot Audio Classification with Attentional Graph Neural Networks. *INTERSPEECH*. 2019, 3649–3653.

[206] Z. Zhang and B. Schuller. Semi-supervised learning helps in sound event classification. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Mar. 2012, 333–336.

[207] S. Zhao, T. Heittola and T. Virtanen. Active Learning for Sound Event Classification by Clustering Unlabeled Data. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, 751–755.

[208] S. Zhao, T. Heittola and T. Virtanen. An Active Learning Method Using Clustering and Committee-Based Sample Selection for Sound Event Classification. *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. 2018, 116–120.

[209] S. Zhao, T. Heittola and T. Virtanen. Active Learning for Sound Event Detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2020), 2895–2905.

[210] X. Zhao, Y. Wang and D. Wang. Robust Speaker Identification in Noisy and Reverberant Conditions. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.4 (2014), 836–845.

[211] X. Zhou, X. Zhuang, M. Liu, H. Tang, M. Hasegawa-Johnson and T. Huang. HMM-based acoustic event detection with AdaBoost feature selection. *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. Ed. by R. Stiefelhagen, R. Bowers and J. Fiscus. Cham, Switzerland: Springer Verlag, 2007, 345–353.

[212] X. Zhuang, X. Zhou, M. A. Hasegawa-Johnson and T. S. Huang. Real-world acoustic event detection. *Pattern Recognition Letters* 31.12 (2010), 1543–1551.

[213] U. Zölzer, ed. *Digital Audio Signal Processing*. 2nd ed. Chichester, UK: John Wiley & Sons, 2008.

PUBLICATIONS

PUBLICATION P1

A. Mesaros, T. Heittola, A. Eronen and T. Virtanen, "Acoustic Event Detection in Real Life Recordings," in *Proceedings of 2010 European Signal Processing Conference*, (Aalborg, Denmark), pp. 1267–1271, 2010.

ACOUSTIC EVENT DETECTION IN REAL LIFE RECORDINGS

Annamaria Mesaros[1], *Toni Heittola*[1], *Antti Eronen*[2], *Tuomas Virtanen*[1]

[1]Department of Signal Processing
Tampere University of Technology
Korkeakoulunkatu 1, 33720, Tampere, Finland
email: annamaria.mesaros@tut.fi, toni.heittola@tut.fi,
tuomas.virtanen@tut.fi

[2]Nokia Research Center
P.O.Box 100, FIN-33721 Tampere, Finland
email: antti.eronen@nokia.com

ABSTRACT

This paper presents a system for acoustic event detection in recordings from real life environments. The events are modeled using a network of hidden Markov models; their size and topology is chosen based on a study of isolated events recognition. We also studied the effect of ambient background noise on event classification performance. On real life recordings, we tested recognition of isolated sound events and event detection. For event detection, the system performs recognition and temporal positioning of a sequence of events. An accuracy of 24% was obtained in classifying isolated sound events into 61 classes. This corresponds to the accuracy of classifying between 61 events when mixed with ambient background noise at 0dB signal-to-noise ratio. In event detection, the system is capable of recognizing almost one third of the events, and the temporal positioning of the events is not correct for 84% of the time.

1. INTRODUCTION

Audio streams, such as broadcast news, meeting recordings, and personal videos contain sounds from a wide variety of sources. Examples include audio events related to human presence, such as speech, laughter, or coughing, or to sounds of animals, objects, nature, or situations. The detection of these events is useful, e.g., for automatic tagging in audio indexing, automatic sound analysis for audio segmentation or audio context classification.

An audio context or scene is characterized by the presence of individual sound events. In this respect, we may want to manage a multi-class description of our audio or video files by detecting the categories of sound events which occur in a file. For example, one may want to tag a holiday recording as being on the ”beach”, playing with the ”children” and the ”dog”, right before the ”storm” came. These are different level annotations, and while the beach as a context could be inferred from acoustic events like waves, wind, and water splashing, the audio events ”dog barking” or ”children” should be explicitly recognized, because such acoustic event may appear in other contexts, too.

The goal of this paper is to present an event detection system for a large and complex dataset. Previous related work includes audio scene recognition [1, 2, 3], analysis of video sound tracks [4, 5], and acoustic event detection [6]. Earlier work commonly considers only a rather limited number of audio events in a small set of audio environments. The work presented in this paper extends the event detection task to a comprehensive set of event-annotated audio material from everyday environments. We consider the task of recognizing and locating audio events in polyphonic long recordings. We use the term ”polyphonic” for denoting recordings in which there are overlapping events, and at one instant of time there is no limitation for the number of event sound sources that can be present.

Our experiments comprise three parts. First, a study of the effect of hidden Markov model (HMM) size and topology for classification performance is performed using a database of isolated audio events. On the same database, we study the effect of the polyphony by adding environmental noise in different signal-to-noise ratios. The environmental noise is selected from a collection of appropriate ambient noises where other similar events can be present to create a realistic polyphonic fragment. Similar classification experiments are also run on real-life recordings, with the purpose of classifying the most prominent audio event in segments of various sizes. The test segments are provided by manual annotation, as it will be explained later. A final experiment is the detection of audio events in long recordings, which includes recognition and temporal positioning of a sequence of events within the recording.

The paper is organized as it follows: Section 2 presents an overview of audio scene recognition and event detection studies we find relevant to our work. Section 3 presents the tests covering isolated sound event classification. Section 4 describes the final choice for the recognition system stucture, the database of real life recordings and the experimental results in classifying and detecting audio events in the recordings. Section 5 presents discussion and conclusions and the orientation towards future work.

2. PREVIOUS WORK

Most of the previous work classifies an audio signal into one of predefined classes using standard features such as mel-frequency cepstral coefficients (MFCC) and classifiers such as hidden Markov models (HMM) or Gaussian mixture models (GMM). In [3], authors compared various features and classifiers in classifying between 24 everyday contexts, such as restaurant, car, library, and office. The system used MFCCs and their first-order time derivatives as features and HMMs with discriminative training for classification. The authors also conducted a listening test to compare the system's performance to the human abilities. The average recognition accuracy of the system was 58%, against 69% obtained in the listening tests, in recognizing between 24 everyday contexts. The accuracies in recognizing six high-level classes were 82% for the system and 88% for the humans.

The work in [7] deals with direct audio context recognition. Individual events are considered to be characteristics of the audio scene, and are not modeled themselves, but included in models of the contexts. The events and contexts are chosen such that to minimize overlapping. The authors present results for classifying 14 different contexts using MFCCs and matching pursuit features, using fixed length segments in training and testing.

In [2], the authors propose unsupervised clustering of interesting events recorded automatically in an office environment. The ”interesting” events are detected by continuous monitoring of background noise and then clustered into discrete categories using unsupervised k-means. Authors of [4] propose a framework for detection of key audio effects in a continuous stream. They use 10 audio effects, distinct enough to be perceived, modeled using HMMs with parameters trained using isolated audio effects from Web, and decode the optimal sequence using the Viterbi algorithm.

Acoustic information is used also for finding interesting segments of video in video content analysis. Authors of [5] present an audio keyword generation system for sports videos based on audio. They use HMMs for classifying semantic events and a support vec-

[1]This work was financially supported by the Academy of Finland.

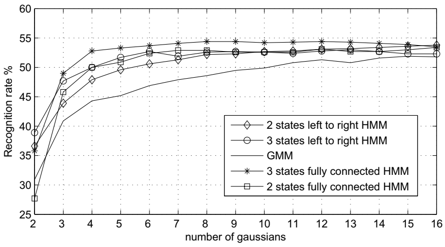


Figure 1: Isolated events classification performance for different size and type models.

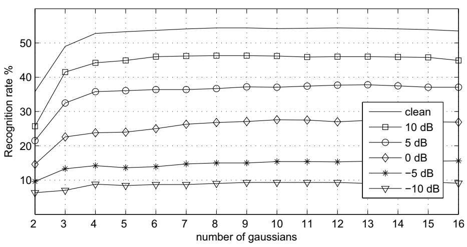


Figure 2: Isolated event classification performance under varying SNR conditions.

tor machine (SVM) classifier for finding audio keywords in soccer, basketball and tennis videos. Audio event detection can find a use also in healthcare monitoring for elderly people [8] or audio-based surveillance [1].

Efforts on acoustic events detection are presented in the CHIL project in their CLEAR evaluation [6]. The goal of the acoustic event detection task is to detect and recognize a closed set of pre-defined acoustic events. The evaluation data consisted of overlapping acoustic events occurring in the CHIL lecture and meeting corpus. Participants to the CLEAR evaluation proposed 5 systems based on HMMs and one on SVMs; the best performing system used HMMs and AdaBoost for feature selection[9]. Our proposal consisted of fully connected HMMs, using MFCCs and optimal path search decoded using the Viterbi algorithm [10].

Despite the research done so far, reliable detection and categorization of audio events from everyday audio is not mature enough for practical applications, such as automatic indexing of video sound tracks. The presented research contributes to the field by presenting a detailed evaluation of an HMM-based event detection system on a realistic and diverse set of audio material.

3. ISOLATED EVENTS CLASSIFICATION

In order to select the appropriate size and type of audio event models, we performed preliminary tests for isolated sound recognition. For this, a collection of isolated sound effects was selected from the Stockmusic online sample database [1], and organized into 61 classes. This database contains a total of 1359 samples belonging to 9 different contexts: crowd, hallway, household, human, nature, office, outdoors, shop, vehicles.

Samples from these classes were randomly selected either to the training set (70%) or to the testing set (30%). The training and testing set randomization was done five times and the average performance was calculated. Isolated event recognition was implemented for the 61 event classes, using MFCC based features and HMMs. We chose the same parametrization method as in [10]. Sixteen MFCCs were extracted from 20 ms long Hamming-windowed frames with 50% frame overlap and 40 mel-bands spanning the frequency range up to the Nyquist frequency were simulated in the frequency domain. The zeroth order coefficient was discarded. In addition to the static MFCC coefficients, we appended the first and second time derivatives. Using these features, an HMM was trained for each audio event class using the Expectation-Maximization (EM) algorithm. In the classification stage, the likelihood of each HMM producing the test observation sequence was obtained using the Viterbi algorithm, and the event was selected as the one corresponding to the HMM giving the largest likelihood.

Figure 1 presents the recognition rates for different size and type of HMMs and number of gaussians per state. At a sufficiently high number of gaussians per state, the system attains its maximum possible performance for the task, which in our case is 54% for 61 events. We also tried adjusting the number of states according to the average length of the audio events; this did not result in higher performance. Most of the fully-connected models became diagonalized during the training. Based on the simulations, it appears that a three-state left-to-right HMM with 4 to 16 mixture densities per state is a good choice for modeling audio events.

We conducted an additional study of how the environment richness influences the recognition of events. To simulate a natural polyphonic environment, we studied the effect of different signal-to-noise ratios, the signal being the event to be recognized and ”noise” being selected from a database of ambient noises [2]. Ambient noise samples were chosen from the same 9 context classes as the sound effects. The background samples were randomly selected for each sound effect from the same context to which the event belongs, and the same background sample was used for the different SNR-cases. The results of sound effects classification under varying SNR conditions is presented in Figure 2 for a three-state HMM as a function of the number of gaussians per state. It can be observed how the performance decreases considerably with the introduced polyphony. This happens also in everyday life; when the acoustic power of the environmental noise is too high compared to individual events, we simply do not hear or recognize them anymore.

4. EVENT DETECTION IN REAL LIFE RECORDINGS

In the event detection in real life recordings, two tasks are evaluated: classification of isolated events in polyphonic recordings and detection of events in continuous sequences. For classification of isolated events, the test data provided to the recognizer consists of a short segment of audio containing one specific event, but the segment can have a rich content meaning that other events may also be present on the duration of the target event to be recognized. This task is similar to the SNR experiments from Section 3. In the acoustic event detection, the system also needs to temporally position the events. The test data consists of an entire track, and the system performs segmentation and classification simultaneously.

4.1 System description

The system for event detection consists of 61 event class models represented by three-state left-to-right HMMs with 16 gaussians per state. The set of features used for constructing the models are the MFCCs. The parameterization was the same as in Section 3.

For event classification, the class corresponding to the model resulting in the largest likelihood for the test observation sequence is chosen as recognition result. For event detection, the 61 models are connected into a network HMM, having equal transition probabilities from one event model to another. The detection task output is an unrestricted sequence of the 61 models, where any model can follow any other and there is no limit for the number of events. The optimal sequence of events is decoded using the Viterbi algorithm. The output of the system contains the timestamps for the recognized

[1] http://stockmusic.com/

[2] http://www.sound-ideas.com/

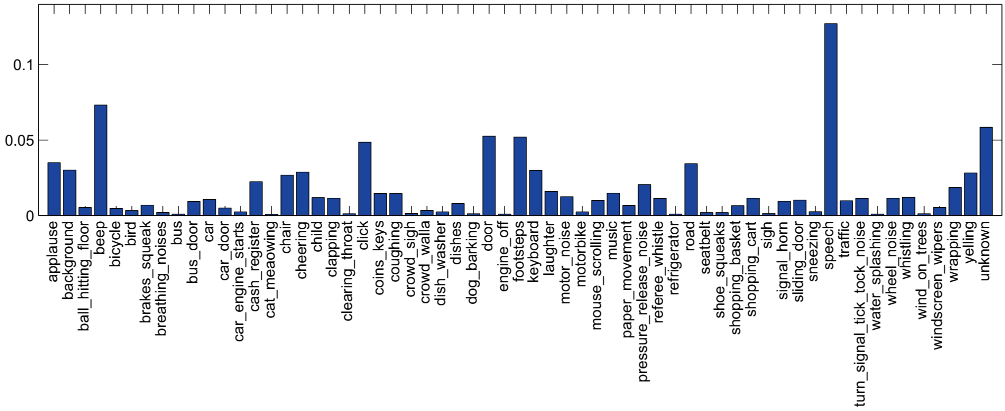


Figure 3: Count-based probabilities for the event classes calculated for the entire database. The histogram is dominated by ”speech”, as it is the most frequently annotated event, appearing in all the recorded contexts.

events, assuming that the system will indicate the most prominent event at a given polyphonic segment.

4.2 Database of real life recordings

For the modeling and recognition of acoustic events we collected long recordings (10 to 30 min each) from ten different acoustic environments (see the list in Table 1). All the recordings are made using a binaural setup, where a person is wearing the microphones in his ears during the recording. The recording equipment consists in a Soundman OKM II Klassik/studio A3 electret microphone and Roland Edirol R-09 wave recorder using 44.1 kHz sampling rate and 24bit resolution.

The events in the recordings were manually annotated by specifying the name and exact location (start and end time) of each audible event within the files. For each context there are 8 to 14 recordings, with a total of 103 recordings in the database. Within each context there are from 9 to 16 annotated event classes, totalling to 61 event classes, and there are many event classes appearing in multiple contexts. We formed distinct classes for events appearing at least 10 times, while more rare events are included in a class labeled as ”unknown”. Figure 3 illustrates the event classes and their frequencies of occurrence within the database. The classes are not balanced, some events are very frequent, while other are very common, as it is expected in a natural environment.

The data was split into non-overlapping training and testing sets such that in five folds all the material gets tested. Individual event instances as annotated are used for training. The features for one event instance were calculated directly from the polyphonic mixture, in the region of each track that was annotated as having that event present. In the case when more events appear simultaneously, the same part of the track (therefore the same observation vectors) was assigned to all the event classes present in that segment. The observations for individual events were used to construct models for each class. Table 1 presents information about the number of event instances extracted from each context.

4.3 Event classification

In this experiment we are interested in recognizing one event per presented test segment, considering that the system will identify the most prominent event in that segment. The experiments were performed in the described five fold setup. In this case, the test data is segmented into chunks containing one event, according to the annotated start and end times for each event instance. These segments are similar to the data used for training the event classes. In this respect, the task is isolated event classification, but with polyphonic audio, where other events may also be present on the duration of the target event to be recognized.

Table 1: Number of events extracted for each context of the recordings

basketball	990	beach	738
bus	1729	car	582
office	1220	hallway	822
restaurant	780	shop	1797
street	827	tracknfield	793

Table 2: Acoustic event classification evaluated using using one, two and three-best list

	one best	2-best	3-best
accuracy	23.8 %	35.4 %	44.1 %

The average recognition accuracy is 23.8%, and some event classes have zero recognition rate. The confusion matrix is presented in Figure 4, and the recognition rates for individual classes are presented in Figure 5. There are cases when one event class is not present both in training and testing, thus we expect it to be wrongly classified, while in other cases there may be acoustic events that are more prominent for a given segment than the target one – for example water splashing is often recognized as wind on trees, which is a concurrent event in the beach recordings. To take into account the possibility of recognizing multiple superimposed events, we chose from one to three best scoring models for each tested file. The results of the experiments are presented in Table 2. The evaluation considers an event to be correctly recognized if its model is among one to three most likely models.

In the SNR experiments from Section 3, the recognition rates drop with approximately 10% every 5 dB. At the 0dB level, the concurrent background ambient noise has the same level as the acoustic event to be classified. At that value, the recognition rate is comparable with the results obtained for the real life recordings. This suggests that the level at which our annotator could still clearly hear

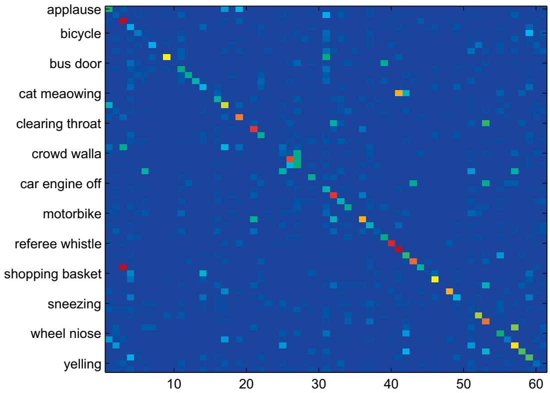


Figure 4: Confusion matrix for event classification. The labels presented in the figure represent every fifth event class in alphabetical order.

and annotate a distinct sound event is when the acoustic power of the power is approximately the same as the power of the event itself.

4.4 Event detection

As mentioned, for the event detection task, the optimal sequence of events is decoded using the Viterbi algorithm within the system HMM network, assuming that the system will indicate the most prominent event at a given time. The output contains the start and end times for the recognized events, marked as the points when the search path goes from one event model to another.

Prior knowledge of the events frequency of occurrence can be used in the detection. This information is presented as a normalized histogram of the event counts, as illustrated in Figure 3. These are prior probabilities for the event classes. The likelihoods of the event classes during recognition will be multiplied by their prior probabilities in order to determine a posterior probability that will then be used in the Viterbi search.

As a performance evaluation measure for the events detection we use the accuracy evaluation metric from the CLEAR 2007 evaluation. This metric is used to score detection of relevant acoustic events (AE). It does not take into account temporal coincidence of the annotated and system output timestamps. It is defined as the F-score (the harmonic mean between precision and recall). In the evaluation, the balanced F-score was used:

$$ACC = 2 * \frac{Precision * Recall}{Precision + Recall},$$

where

$$Precision = \frac{\text{number of correct system output AEs}}{\text{number of all system output AEs}}$$

and

$$Recall = \frac{\text{number of correctly detected reference AEs}}{\text{number of all reference AEs}}$$

Table 3: Acoustic event detection evaluation results

system	Precision	Recall	Accuracy
no priors	38.9%	24.5%	30.1%
using priors	39.6%	24.2%	30.0%

Table 4: Acoustic event detection error

system	missed events	false alarms	substitutions	overall error
no priors	60.6%	1.4%	22.1%	84.1%
using priors	60.7%	1.4%	21.8%	84.0%

The system output is considered correct if there exists at least one annotated sound event whose temporal centre is situated between the timestamps of the system output, and the annotated label and system output are similar, or if the temporal centre of the system output lies between the timestamps of at least one annotated event and the annotated label and system output are similar. The annotated sound event is considered correctly detected if there exists at least one system output whose temporal centre is situated between the timestamps of annotated sound event and the labels are similar, or if the temporal centre of the annotated sound event lies between the timestamps of at least one system output and the labels are similar. The results are presented in Table 3.

The temporal resolution of the detected acoustic events is scored using the metric for Speaker Diarization, adapted to the task of audio event detection in the CLEAR evaluation. A one-to-one mapping of the reference acoustic events to the acoustic events output by the system is computed, and the measure is the aggregation over all reference acoustic events of the time that is jointly attributed to both the reference and the corresponding system output acoustic event to which that reference events are mapped. This is computed over all audio segments, including regions of overlapping.

The overall error score ER will be computed as the fraction of the time that is not attributed correctly to an acoustic event:

$$ER = \frac{\sum_{seg}\{dur(seg) * max(N_{ref}, N_{sys}) - N_{correct}\}}{\sum_{seg}\{dur(seg) * N_{ref}\}}$$

where the audio data is divided into adjacent segments whose border coincide with the points where either a reference or a system output acoustic event starts or stops, so that for the given segment, the number of current reference AEs and the number of system output AEs do not change. For each segment *seg*, *dur* is the duration of the *seg*, N_{ref} is the number of reference AEs in *seg*, N_{sys} is the number of system output AEs in *seg* and $N_{correct}$ is the number of reference AEs in *seg* which have a corresponding mapped system output AEs in *seg* .

The overall detection error of the system and some details about the errors are presented in Table 4. The total amount of scored time is 920 min; this represents the added duration of all annotated events, being 2.5 times more than the actual time covered by overlapping events. The overall acoustic event detection error of the presented system for the 61 event classes is 84.1% of the total scored time.

Using the prior information based on overall events counts did not improve the results for event detection. Such direct count may not reflect the true probability of events in different contexts; because of averaging over all the contexts, the histogram in Figure 3 is dominated by ”speech”. Indeed, speech is present in all the contexts and it overlaps practically all other events, and also gets a lot of confusions in the classification.

In the audio events detection of the CLEAR evaluation, the best system score was 36.3% accuracy and 99.5% detection error. In comparison, our system has a lower detection error for a much higher number of classes, but the accuracy of recognition is lower.

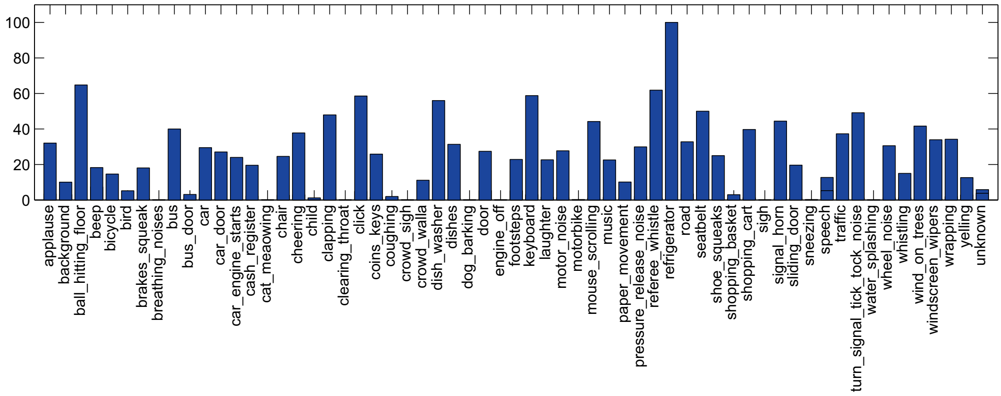


Figure 5: Event classification performance of individual classes

.

5. CONCLUSIONS

This paper presented a detailed evaluation of an HMM-based event detection and classification system using recordings of ten different natural environments. Three different tests were performed. A study of the topology and size of the selected models was performed on a database containing isolated audio events, obtaining a maximum performance of 54% for the three-state left-to right and fully-connected HMMs. Based on these results, we selected a three-state left-to-right model for the subsequent experiments. We performed a similar event classification task on the real-life recordings, obtaining a recognition performance of 24%. Similar performance was obtained in isolated events recognition with with background noise mixed at 0 db SNR, suggesting that this is the level where humans can clearly hear and annotate an audio event in a natural context. For detecting successive events in a long recording, the proposed system has an accuracy of 30% for 61 classes and a detection error of 84.1%. Using prior information based on overall event count did not bring any improvement. We think this is due to adding up all the events from different environments, which averages out the differences in count between events specific to certain environments. Our future work will consider e.g. using missing feature techniques for improving the event detection robustness in polyphonic mixtures. The current event detection system is used in an audio context recognition system based on acoustic events.

REFERENCES

[1] C. Clavel, T. Ehrette, and G. Richard, “Events detection for an audio-based surveillance system,” in *Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, ICME 2005, July 6-9, 2005, Amsterdam, The Netherlands*, 2005, pp. 1306–1309.

[2] A. Härmä, M. F. McKinney, and J. Skowronek, “Automatic surveillance of the acoustic activity in our living environment,” in *IEEE International Conference onMultimedia and Expo*, Los Alamitos, CA, USA, 2005, vol. 0, p. 4 pp., IEEE Computer Society.

[3] A.J. Eronen, V.T. Peltonen, J.T. Tuomi, A.P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, “Audio-based context recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, Jan. 2006.

[4] R. Cai, L. Lu, A. Hanjalic, H-J. Zhang, and L-H. Cai, “A flexible framework for key audio effects detection and auditory context inference,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 1026–1039, 2006.

[5] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, “Audio keywords generation for sports video analysis,” *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 4, no. 2, pp. 1–23, 2008.

[6] Rainer Stiefelhagen, Rachel Bowers, and Jonathan Fiscus, Eds., *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, Springer-Verlag, Berlin, Heidelberg, 2008.

[7] S. Chu, S. Narayanan, and C-C. Jay Kuo, “Environmental sound recognition with time-frequency audio features,” *IEEE Transactions on Speech, Audio, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

[8] Ya-Ti Peng, Ching-Yung Lin, Ming-Ting Sun, and Kun-Cheng Tsai, “Healthcare audio event classification using hidden markov models and hierarchical hidden markov models,” in *ICME’09: Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, Piscataway, NJ, USA, 2009, pp. 1218–1221, IEEE Press.

[9] X. Zhou, X. Zhuang, M. Liu, H. Tang, M. Hasegawa-Johnson, and T. Huang, “Hmm-based acoustic event detection with adaboost feature selection,” in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, Berlin, Heidelberg, 2008, pp. 345–353, Springer-Verlag.

[10] T. Heittola and A. Klapuri, “TUT acoustic event detection system 2007,” in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, Berlin, Heidelberg, 2008, pp. 364–370, Springer-Verlag.

PUBLICATION P2

T. Heittola, A. Mesaros, A. Eronen and T. Virtanen, “Context-Dependent Sound Event Detection,” in *EURASIP Journal on Audio, Speech and Music Processing*, Vol. 2013, No. 1, 13 pages, 2013

EURASIP Journal on
Audio, Speech, and Music Processing
a SpringerOpen Journal

RESEARCH Open Access

Context-dependent sound event detection

Toni Heittola[1]*, Annamaria Mesaros[1], Antti Eronen[2] and Tuomas Virtanen[1]

Abstract

The work presented in this article studies how the context information can be used in the automatic sound event detection process, and how the detection system can benefit from such information. Humans are using context information to make more accurate predictions about the sound events and ruling out unlikely events given the context. We propose a similar utilization of context information in the automatic sound event detection process. The proposed approach is composed of two stages: automatic context recognition stage and sound event detection stage. Contexts are modeled using Gaussian mixture models and sound events are modeled using three-state left-to-right hidden Markov models. In the first stage, audio context of the tested signal is recognized. Based on the recognized context, a context-specific set of sound event classes is selected for the sound event detection stage. The event detection stage also uses context-dependent acoustic models and count-based event priors. Two alternative event detection approaches are studied. In the first one, a monophonic event sequence is outputted by detecting the most prominent sound event at each time instance using Viterbi decoding. The second approach introduces a new method for producing polyphonic event sequence by detecting multiple overlapping sound events using multiple restricted Viterbi passes. A new metric is introduced to evaluate the sound event detection performance with various level of polyphony. This combines the detection accuracy and coarse time-resolution error into one metric, making the comparison of the performance of detection algorithms simpler. The two-step approach was found to improve the results substantially compared to the context-independent baseline system. In the block-level, the detection accuracy can be almost doubled by using the proposed context-dependent event detection.

1 Introduction

Sound events are good descriptors for an auditory scene, as they help describing and understanding the human and social activities. A *sound event* is a label that people would use to describe a recognizable event in a region of the sound. Such a label usually allows people to understand the concept behind it and associate this event with other known events. Sound events can be used to represent a scene in a symbolic way, e.g., an auditory scene on a busy street contains events of passing cars, car horns, and footsteps of people rushing. Auditory scenes can be described with different level descriptors to represent the general context (street) and the characteristic sound events (car, car horn, and footsteps). As a general definition, a *context* is information that characterizes the situation of a person, place, or object [1]. In this study, the definition of context is narrowed to the location of auditory scene.

Automatic sound event detection aims at processing the continuous acoustic signal and converting it into such symbolic descriptions of the corresponding sound events present at the auditory scene. The research field studying this process is called computational auditory scene analysis [2]. Automatic sound event detection can be utilized in a variety of applications, including context-based indexing and retrieval in multimedia databases [3,4], unobtrusive monitoring in health care [5], surveillance [6], and military applications [7]. The symbolic information about the sound events can be used in other research areas, e.g., audio context recognition [8,9], automatic tagging [10], and audio segmentation [11].

Our everyday auditory scenes are usually complex in sound events, having a high degree of overlapping between the sound events. Humans can easily process this into distinct and interpreted sound events, and follow a specific sound source while ignoring or simply acknowledging the others. This process is called auditory scene analysis [12]. For example, one can follow a conversation in a busy background consisting of other people talking. Human sound perception is also robust to many

*Correspondence: toni.heittola@tut.fi
[1] Department of Signal Processing, Tampere University of Technology, P.O. Box 553, Tampere, Finland
Full list of author information is available at the end of the article

environmental conditions influencing the audio signal. Humans can recognize the sound of footsteps, regardless of whether they hear footsteps on a pavement or on gravel, in the rain or in a hallway. In case of an unknown sound event, humans are able to hypothesize as to the source of the event. Humans use their knowledge of the context to predict which sound events they are likely to hear, and to discard interpretations that are unlikely given the context [13]. In real-world environments, sound events are related to other events inside a particular environment, providing a rich collection of contextual associations [14]. In the listening experiments, this facilitatory effect of the context to the human sound identification process has been found to partly influence the perception of the sound [15].

Automatic sound event detection systems are usually designed for specific tasks or specific environments. There are a number of challenges in extending the detection system to handle multiple environments and a large set of events. Event categories and variance within each category make the automatic sound event recognition problem difficult even with well-represented categories when having clean and undistorted signals. The overlapping sound events that constitute a natural auditory scene create an acoustic mixture signal that is more difficult to handle. Another challenge is the presence of certain sound events in multiple contexts (e.g., footsteps present in contexts like street, hallway, beach) calling for rules in modeling of the contexts. Some events are context specific (e.g., keyboard sounds present in the office context) and their variability is lower, as they always appear in similar conditions.

A possible solution to these challenges is to use the knowledge about the context in the sound event detection in the same manner as humans do [15], by reducing the search space for the sound event based on the context. We achieve this by implementing a first stage for audio context recognition and event set selection. The context information will provide rules for selecting a certain set of events. For example, it will determine excluding the footsteps class when the tested recording is from inside a car. A smaller set of event models will reduce the complexity of the event detection stage and will also limit the possible confusions and misclassifications. Further, context-dependent prior probabilities for events can be used to predict most likely events for the given context. The context information offers also possibilities for improving the acoustic sound event models used in the detection system. A context-dependent training and testing has the benefit of better fitting acoustic models for the sound event classes, by using only examples from a given context. For example, footsteps are acoustically different on a corridor (hallway context) than on the sand (beach context), and using specific models should be beneficial.

This article studies how to use context information in the sound event detection process, and how this additional information improves the detection accuracy. The proposed sound event detection system is composed of two stages: a context recognition stage and a sound event detection stage. Based on the recognized context, a context-specific set of sound events is selected for the sound event detection stage. In the detection stage, context-dependent acoustic models and count-based event priors are used. Two alternative event detection approaches are studied. In the first one, monophonic event sequence is outputted by detecting most prominent sound event at each time instance. In the second approach, a polyphonic event sequence is produced by detecting multiple overlapping sound events.

The rest of this article is organized as follows. Section 2 discusses related previous work, and Section 3 explains basic concepts of sound event detection. Section 4 presents a detailed description of the proposed context-dependent sound event detection system. Section 5 presents the audio database and metrics used in the evaluations. Section 6 contains detailed results of the evaluations and the discussions of the results. Finally, concluding remarks and future research directions are given in Section 7.

2 Previous work

Early research related to the classification of sounds for everyday life has been concentrating on problems with specific sounds. Examples include gunshots [16], vehicles [17], machines [18], and birds [19]. In addition to this, usually a low number of sound categories are involved in the studies, specifically chosen to minimize overlapping between different categories, and evaluations are carried out with one or very small set of audio contexts (kitchen [20], bathroom [21], meeting room [22], office and canteen [23]). Many of these previously presented methods are not applicable as such for the automatic sound event detection for continuous audio in real-world situations.

The problem of sound event detection in real environments having a large set of overlapping events was addressed in the acoustic event detection task (AED) of the Classification of Events, Activities and Relationship (CLEAR) evaluation campaign [24]. The goal of the AED task was to detect non-speech events in the meeting room environment. The metric used in the evaluation was designed for the detection system outputting a monophonic sequence of sound events. The best performing system submitted to the evaluation achieved a 30% detection accuracy by using AdaBoost-based feature selection and a Hidden Markov Model (HMM) classifier [25]. Later this study was extended into a two-stage system having a tandem connectionionist-HMM-based classification stage and a re-scoring stage [26]. The authors achieved a 45% detection accuracy on the CLEAR evaluation database. Sound event detection for a wider set of real-world audio contexts was studied in [27]. A system based on Mel-frequency cepstral coefficients (MFCC) features and an HMM classifier achieved on average a 30% detection accuracy over ten real-world audio contexts.

In addition to the acoustic features and classification schemes, different methods have been studied to include prior knowledge of the events to the detection process. Acoustically homogeneous segments for the environment classification can be defined using frame level n-grams, where n-grams are used to model the prior probabilities of frames based on previously observed ones [28]. In a complex acoustic environment with many overlapping events, the number of possible combinations is too high to be able to define such acoustically homogeneous segments and for modeling transitions between them. In [3], a hierarchical probabilistic model was proposed for detecting key sound effects and audio scene categories. The sound effects were modeled with HMMs, and a higher-level model was used to connect individual sound effect models through a grammar network similar to language models in speech recognition. A method of modeling overlapping event priors has been addressed in [29], by using probabilistic latent semantic analysis to calculate priors and learn associations between sound events. The context-recognition stage proposed in this article will solve the associations of the sound events by splitting the event set into subsets according to the context. Furthermore, the count-based priors estimated from training material can be used to provide probability distributions for the sound events inside each context.

In order to be able to do context-dependent sound event detection, we introduce a context recognition step. In recent years, there has been some research on modeling what is called *context awareness* in sound recognition. One group of studies focuses on estimating the context of an audio segment with varying classification techniques [8,30,31]. In these studies the context is represented by a class of sounds that can be heard in some type of environment, such as cars at a street, or people talking in a restaurant. Depending on the number of context classes that are learned, the recognition rates of these methods vary between 58 (24 classes, [30]) and 84% (14 classes, [8]). Although these results are promising, the methods that are used have some attributes that make them less suitable for automatic sound event detection. Features that are used to classify an audio interval are assumed to represent information that is specific for a class, and therefore, the context class to which an audio interval belongs gives primarily information about its acoustic properties. Tasks in multimedia applications (or a comparable setup in environmental sound classification, as in [8]) generally entail that a small audio interval, typically not longer than a few seconds, is classified as a sample of one context out of a dataset with a limited set of distinct contexts, which are stored as a collection of audio files. A second group of studies on context awareness addresses some of the above issues by retrieving semantic relatedness of sound intervals rather than the similarity of their acoustic properties [32,33]. For example, in [32] the intervals are clustered based on the similarity. Our approach for event detection will include a step of context recognition by classifying short intervals, before the main step of event detection.

3 Event detection

This section explains the sound event detection approach used in the proposed method, which recognizes and temporally locates sound events in recordings. In Section 4, this approach is extended to use context-dependent information.

3.1 Event models

The coarse shape of the power spectrum of the recording from the auditory scene is represented with MFCCs. They provide a good discriminative performance with reasonable noise robustness. In addition to the static coefficients, their first and second time derivatives are used to describe the dynamic properties of the cepstrum.

Sound-event-conditional feature distributions are modeled using continuous-density HMMs. Left-to-right model topology having three states was chosen to represent sound events having a beginning, a sustained part, and an end part. A mixture of multivariate Gaussian density functions is used in modeling the probability density functions of observations in each state. The acoustic models are trained using audio signals where the start and end times of events as well as their classes have manually been annotated. The traditional approach would be to use non-overlapping sound events to train the acoustic event models. However, realistic auditory scenes are usually too complex to provide enough such material for reliable training. Thus, each event instance annotated represents one training sample for the model of the event class regardless whether there were overlapping events present or not. The regions of the sound that contain overlapping events are used as training instances of both event classes when training the models. The assumption behind this procedure is that in the model training stage the variability caused by overlapping sound events classes will average out and the models will learn a reliable representation of the target sound events. The procedure of assigning training material to the event classes is illustrated in Figure 1. The models for sound events are trained with these samples using the Baum–Welch algorithm [34].

In the testing stage, the sound event models are connected into a network with transitions from each model to any other. A model network is shown in Figure 2.

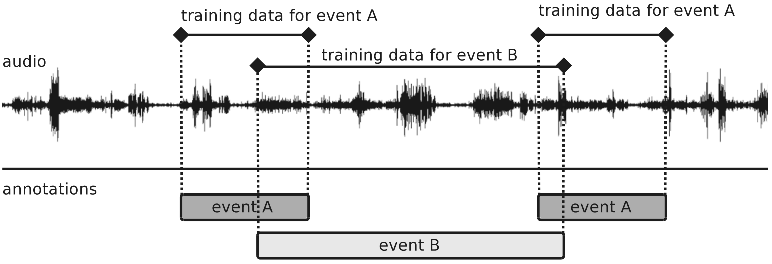

training data for event A
training data for event A
audio
training data for event B
annotations
event A
event A
event B

Figure 1 Training material containing overlapping sound events is used to train both sound event models.

Since it is possible that a test recording will contain some sound events which were not present in the training set, the system has to be able deal with such situations. A universal background model (UBM) is often used in speaker recognition to capture general properties of the signal [35]. We are using a UBM to capture events which are unknown to the system. A one-state HMM is trained with all available recordings for this purpose.

3.2 Count-based priors

Equally probable events can be represented by a network with equal inter-model transition probabilities. In this case, the output will be an unrestricted sequence of relevant labels, in which any event can follow any other.

In reality, the sound events are not uniformly distributed. Some events are more likely than others, e.g., speech is more common than a car alarm sound. If we regard each event as a separate entity and model the event counts, the histogram of the event counts inside certain context will provide us event priors. The event priors can be used to control event transitions inside the sound event model network shown in Figure 2. The count-based event priors are estimated from the annotated training material.

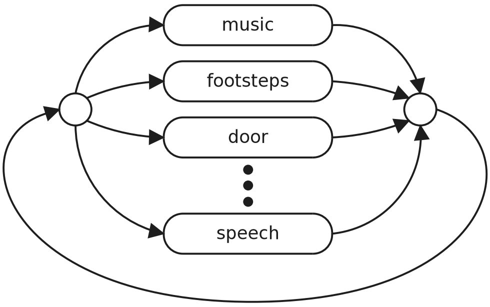


Figure 2 Fully connected sound event model network.

3.3 Detection

We will present two alternative approaches for the sound event detection: in the first one, we find the most prominent event at each time instance, and in the second one we find a predefined number of overlapping events. The detection of the most prominent event will produce a monophonic event sequence as an output. This approach is later referred as *monophonic* detection. The detection of overlapping events will produce a polyphonic event sequence as an output. This approach is later referred as *polyphonic* detection. Examples of the outputs of these two approaches are shown in Figure 3.

3.3.1 *Monophonic detection*

Segmentation of a recording into regions containing the most prominent event at a time will be obtained by doing Viterbi decoding [36] inside the network of sound event models. Transitions between models in this network are controlled by event prior probabilities. The balance between the event priors and the acoustic model is adjusted using a weight in combining the two likelihoods when calculating the path cost through the model network. A second parameter, insertion penalty, controls the number of events in the event sequence by controlling the cost of inter-event transition. These parameters are experimentally chosen using a development set.

3.3.2 *Polyphonic detection*

As discussed in Section 2, the previous studies related to sound event detection consider audio scenes with overlapping events that are explicitly annotated, but the detection results are presented as a sequence that is assumed to contain only the most prominent event at each time. In this respect, the systems output only one event at each time, and the evaluation considers the output correct if the detected event is one of the annotated ones. The performance of such systems is very limited in the case of rich multisource environments.

In order to detect overlapping events, we propose to use consecutive passes of the Viterbi algorithm as proposed in [37] for the detection of overlapping musical

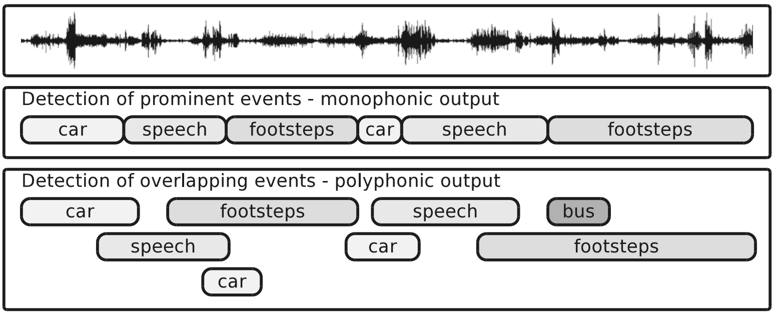

Detection of prominent events - monophonic output
car
speech
footsteps
car
speech
footsteps
Detection of overlapping events - polyphonic output
car
footsteps
speech
bus
speech
car
footsteps
car

Figure 3 Example of sound event detection output for two approaches: monophonic system output and polyphonic system output.

notes. After one iteration, the decoded path through the model network is marked and the next iteration is prohibited from entering any states belonging to the sound event decoded at that frame in the previous iteration. The UBM is allowed in each iteration. This method will provide iterative decoding of the next-best path containing events that are at each time different than in the previously decoded one. This is difficult to achieve with conventional N-best decoding, which provides too many paths that have only minor state changes between them. These state changes do not produce the desired outcome. The proposed approach is illustrated in Figure 4. The number of iterations is chosen depending on the expected polyphony of the acoustic material.

4 Context-dependent event detection

Many sound events are acoustically dissimilar across contexts, and in these cases usage of context-specific acoustic models should provide better modelling accuracy. Sound events also have context-dependent prior probabilities, and using more accurate prior probabilities should also increase detection accuracy. Thus, we propose a sound event detection system utilizing the context information. The proposed system has two stages. In the first stage, the recording is tested for audio context classification. The second stage is the event detection. Based on the recognized context label, a specific set of sound event models is selected and acoustics models trained with the context-dependent material are selected to be used in the detection stage. In addition to this, context-dependent event priors are applied in the event detection. The system overview is presented in Figure 5. The details of each stage will be presented in the following sections.

4.1 Context recognition

As discussed in Section 2, an audio context can be recognized robustly among a small and restricted set of context classes. For our system, we chose a simple state-of-the-art context recognition approach [30] based on MFCCs and Gaussian mixture models (GMMs).

In the recognition stage, the audio is segmented into 4-second segments which are classified individually using the context models. Log-likelihoods are accumulated over all the segments and the model with the highest total

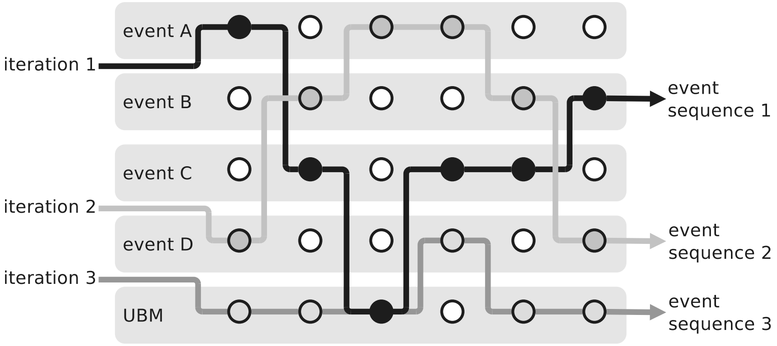


Figure 4 Concept of multiple path decoding using three consecutive passes of Viterbi algorithm.

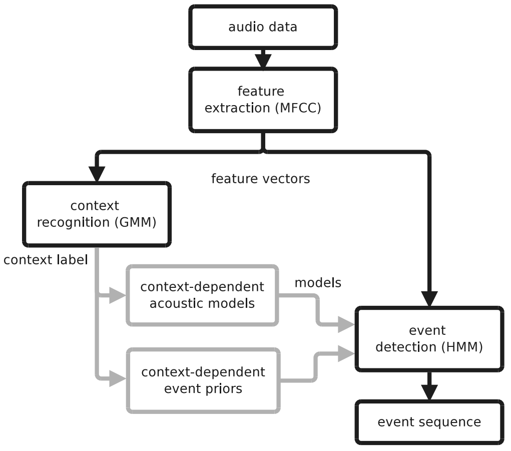

audio data
feature extraction (MFCC)
feature vectors
context recognition (GMM)
context label
context-dependent acoustic models
models
context-dependent event priors
event detection (HMM)
event sequence

Figure 5 System overview.

likelihood is given as the label for the recording. The performance of context recognition will influence the performance of the sound event detection, as incorrectly recognized context will lead to choosing a wrong set of events for the event detection stage. Results for the context recognition are presented in Section 6.1.

The context models used in the context recognition stage are essentially identical to the context-dependent UBMs later used in the event detection stage. This simplifies the training process of the whole system and speeds up the event detection process allowing the calculated observation probabilities to be shared between stages.

4.2 Context-dependent modeling

In order to have more accurate modeling, the acoustic models for sound events are trained within each available context. Context-dependent count-based priors for the sound events are collected from the annotations of training material.

In the testing stage, the set of possible sound events is determined by the context label provided by the context recognition stage. The sound event models belonging to the recognized context will be selected and connected into a network with transitions from each model to any other (see Figure 2). The transitions between events are controlled with count-based event priors estimated for the recognized context.

5 Evaluation setup

The sound event detection system was trained and tested using an audio database collected from various contexts. The system was evaluated using an established evaluation metric [38] and a new metric introduced for a better understanding of the overlapping event detection results.

5.1 Database description

A comprehensive audio database is essential for training context and sound event models and for estimating count-based event priors. To the best of the authors' knowledge, there are only two publicly available audio databases for sound event detection from auditory scene. The database used in CLEAR 2007 evaluation [38] contains only material from meeting rooms. The DARES-G1 database [39] published in 2009 offers a more diverse set of audio recordings from many audio contexts. Event annotations for this database have been implemented using free-from event labels. The annotations would require label grouping in order to make the database usable for the event detection. At the time of this study, there was not any multi-context database publicly available that could be used for the evaluation without additional processing, and we recorded and annotated our own audio database. Our aim was to record material from common everyday contexts and to have as representative collection of audio scenes as possible.

The recordings for the database were collected from ten audio contexts: basketball game, beach, inside a bus, inside a car, hallways, inside an office facility, restaurant, grocery shop, street, and stadium with track and field events. Hallways and office facility contexts were selected to represent typical office work environments. The street, bus, and car contexts represent typical transportation scenarios. The grocery shop and restaurant contexts represent typical public space scenarios, whereas the beach, basketball game, and track and fields event contexts represent examples of leisure time scenarios.

The database consists of 103 recordings, each of which is 10–30-min long. The total duration of recordings is 1133 min. Each context is represented by 8 to 14 recordings. The material for the database was gathered using a binaural audio recording setup, where a person is wearing the microphones in his/her ears during the recording. The recording equipment consists of a Soundman OKM II Klassik/Studio A3 electret microphone and a Roland Edirol R-09 digital recorder. Recordings were done using 44.1 kHz sampling rate and 24-bit resolution. In this study, we are using monophonic versions of the recordings, i.e., two channels are averaged to one channel.

The recordings are manually annotated indicating the start and end times of all clearly audible sound events in the auditory scene. Annotations were done by the same person responsible of the recordings; this ensured as detailed as possible annotations since the annotator had prior knowledge of the auditory scene. In order to help the annotation of complex contexts, like street, also a low-quality video was captured during the recording of audio to help the annotator recall the auditory scene while doing annotation. The annotator had the freedom to choose descriptor labels for the sound events. The event labels used in the annotations were manually grouped into 61 distinct event classes. Grouping was done by combining labels describing essentially the same sound event, e.g., "cheer" and "cheering", or labels describing acoustically very similar event, e.g., a "barcode reader beep" and a "card reader beep". Event classes were formed from events appearing at least ten times within the database. More rare events were included in a single class labeled as "unknown".

Figure 6 illustrates the event classes and their frequencies of occurrence for different contexts in the database. Each context contains 9 to 16 event classes and many event classes appear in multiple contexts (e.g., speech). There are also event classes which are highly context specific (e.g., dishes, or referee whistle). As expected in a natural auditory scenes, the event classes are not well balanced. It can been seen that some events are context specific (e.g., pressure release noise in the bus context), while others are very common across different contexts (e.g., speech). The number of events annotated per context is presented in Table 1.

5.2 Performance evaluation

In order to provide comparable metrics to the previous studies [25-27], in the performance evaluations we are using two metrics also used in the CLEAR 2007 evaluation [38]. The CLEAR evaluation defines the calculation of the precision and recall for the event detection, and the balanced F-score is calculated based on these. This accuracy metric is later denoted by ACC. The CLEAR evaluation also defines a temporal resolution error to represent the erroneously attributed time. This metric is later denoted by ER. Exact definition of these metrics can be found in the evaluation guidelines [38].

For evaluating a system output with overlapping events, the recall calculated in this way is limited by the number of events the system can output, compared to the number of events that are annotated. As a consequence, even if the output contains only correct events, the accuracy for the event detection is limited by the used metric. The temporal resolution error represents all the erroneously attributed time, including events wrongly recognized and events missed altogether by the lack of sufficient polyphony in the detection. The two metrics are therefore complementary, and tied to the polyphony of the annotation. This complicates the optimization of the event detection system into finding a good balance between the two.

In order to tackle this problem and to have a single understandable metric for sound event detection, we propose a block-wise detection accuracy metric. The metric combines the correctness of the event detection with a coarse temporal resolution determined by the length of the block used in the evaluation.

The proposed block-wise metric will evaluate how well the events detected in non-overlapping time blocks coincide with the annotations. The detected events are regarded only at the block level. In the evaluations, we are using two block lengths: 1 (later denoted by A1) and 30 s (later denoted by A30). This metric is designed

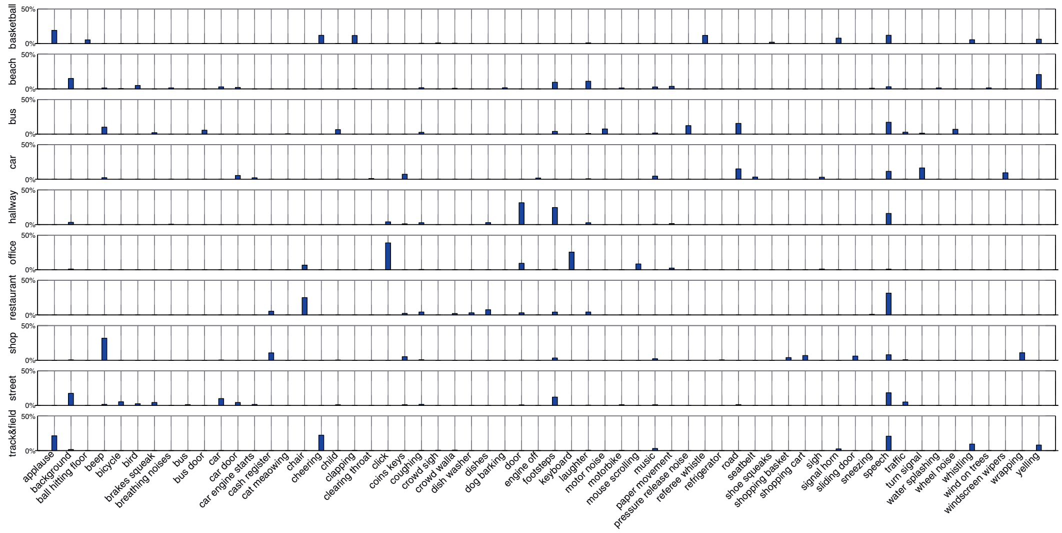


Figure 6 Percentage of sound event classes annotated per audio context in the database.

Table 1 Number of events annotated per context and total length of recordings (in minutes)

	Number of events	Length
Basketball game	990	80
Beach	738	197
Inside a bus	1729	146
Inside a car	582	111
Office facility	1220	105
Hallway	822	100
Restaurant	780	96
Grocery shop	1797	88
Street	827	102
Track & field stadium	793	108

for applications requiring a fairly coarse time resolution, placing more importance into finding the correct events within the block than finding their exact location. Inside the blocks, we calculate precision and recall. Precision is defined as the number of correctly detected sound event classes divided by the total number of event classes detected within the block. Recall is defined as the number of correctly detected sound event classes divided by the number of all annotated event classes within the block. An event is regarded as correctly detected if it has been detected somewhere within the block and the same event label also appears in the annotations within the same block. The accuracy represented by the *F*-score is calculated based on the precision and recall by the formula:

$$\text{Block accuracy} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

where

$$\text{Precision} = \frac{\text{Number of correct events}}{\text{Number of detected events}} \tag{2}$$

and

$$\text{Recall} = \frac{\text{Number of correctly detected events}}{\text{Number of all annotated events}}. \tag{3}$$

An illustration of how this metric works can be seen in Figure 7. In block 1, the annotated events are A, B, C, and D. The monophonic system output for the block 1 contains events A, C, and E. The events A and C are correctly detected by the system. For this block, the precision is 2 out of 3 (2/3) and recall 2 is out of 4 (2/4). The calculated block-wise accuracy for this block is 57.1% and the average block-wise accuracy for the entire example is 57.3%. For comparison, the CLEAR metrics are calculated on the level of entire output. The detection accuracy (ACC) is 76.2% having a precision 8/12 and recall 8/9. The time resolution error (ER) is calculated by counting the units that are wrongly labeled or missed altogether (42) and dividing it with the total number of units (51) covered by the annotated events. This results in a 82.4% time resolution error.

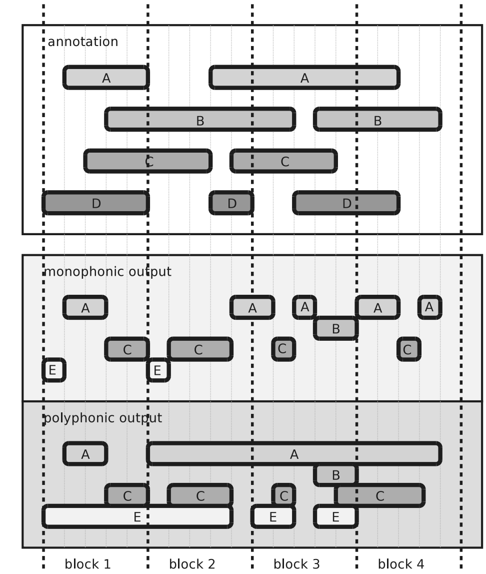


Figure 7 Block-wise accuracy for sound event detection.

For the polyphonic system output, the block-wise accuracy for the first block is 57.1% and the average accuracy for the entire example 58.3%. This is easily comparable with the same metric for the monophonic output. The CLEAR metric for the detection accuracy (ACC) is 63.2% (precision 6/10 and recall 6/9). The time resolution error (ER) is 109.8%, having 56 wrongly labeled or missed time units, compared to 51 in the annotation. This makes it hard to compare the monophonic and polyphonic outputs. In addition to this, an error value over 100% does not have proper interpretation. The proposed block-wise metric is comparable among monophonic and polyphonic outputs, with similar accuracy in the two illustrated cases. Therefore, the metric is equally valid for a system outputting only one event at time (monophonic output) as for a system outputting overlapping events (polyphonic output).

6 Experimental results

The database was split randomly into five equal-sized file sets, with one set being used as test data and other four for training the system. The split was done five times for a fivefold cross-validation setup. One fold was used in the development stage for determining parameters in the decoding. The evaluation results are presented as the average of the other four folds.

Table 2 Context recognition results

	4 s	20 s	40 s	Whole signal
Overall	**70.0**	**80.7**	**85.0**	**91.0**
Context-wise results				
Basketball	91.0	99.0	100.0	100.0
Beach	57.0	69.0	71.0	81.0
Bus	41.0	52.0	58.0	67.0
Car	84.0	93.0	95.0	100.0
Hallway	55.0	60.0	67.0	75.0
Office	85.0	87.0	88.0	88.0
Restaurant	77.0	89.0	95.0	100.0
Shop	72.0	87.0	94.0	100.0
Street	52.0	76.0	83.0	100.0
Track&Field	86.0	96.0	98.0	100.0

Percentage of correctly recognized segments.

Both the context recognition stage and the event detection stage used MFCC features and shared the same parameter set. MFCCs were calculated in 20-ms windows with a 50% overlap from the outputs of a 40-channel filterbank which occupied the frequencies from 30 Hz to half the sampling rate. In addition to the 16 static coefficients, the first and second time derivatives were also used.

In the event detection stage, the parameters controlling the balance between the event priors, the acoustic model, and the sequence length were experimentally chosen using a development set by finding parameter values which resulted in an output comprising approximately the same total amount of sound events that was manually annotated for the recording.

6.1 Context recognition

Context recognition was performed using the method presented in Section 4.1. The number of Gaussian distributions in the GMM model was fixed to 32 for each context class. This amount of Gaussian distributions was found to give a good compromise between computational complexity and recognition performance in the preliminary studies conducted with the development set.

The performance of the context recognition is presented in Table 2 as a fourfold average performance for the evaluation sets for four different segment lengths: 4 s, 20 s, 40 s, and the whole signal. Figure 8 shows the context recognition performance as a function of segment length used in the recognition. It can be seen that already after 2–3 min the system achieves a good recognition accuracy. A decision about the context could be taken already after the first minutes, in order to minimize the complexity of the context recognition stage and avoid processing the whole recording. However, we use the decision obtained after processing the whole signal to maximize the recognition accuracy. When using the whole length of the recording for the decision, six out of ten contexts have perfect 100% recognition rate, and rest of the contexts have also reasonable good, around 80% recognition rate.

The performance could positively be affected by the fact that recordings for the same context were done around the same geographical location, e.g., along the same street. Thus, the training and testing sets might contain recordings around the same area having quite a similar auditory scene, leading to over-optimistic performance.

6.2 Monophonic event detection

First we study the accuracy of the proposed system to find the most prominent event at each time instance. Since the performance of the context recognition stage affects on the selected event set for the event detection, the system is first tested when provided with the ground-truth context label. This will provide us the maximum attainable performance of the monophonic event detection. Later the system is evaluated in conjunction with the context recognition stage to provide a realistic performance evaluation. The system is evaluated using either uniform event priors or count-based event priors.

The number of Gaussian distributions per state in the sound event models was fixed to 16 for each event class.

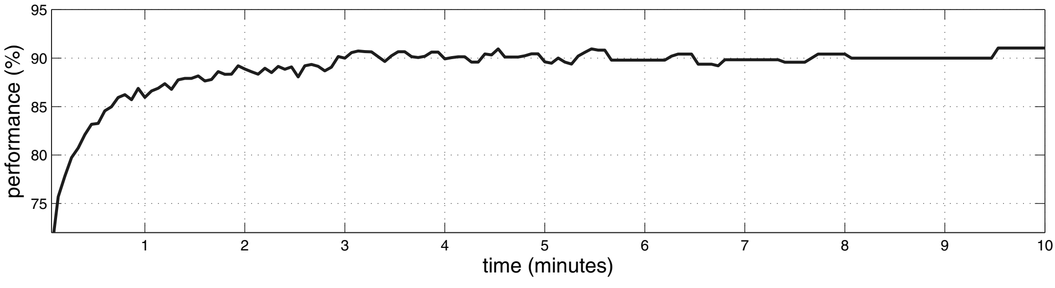


Figure 8 Context recognition performance as a function of used segment length.

This was found to give a high enough accuracy in the preliminary studies using the development set.

All the results are calculated as an average of the four test sets. The results are evaluated first with the CLEAR metrics (ACC and ER) in order to provide a way to compare the results to those of previously published systems [27,29]. In addition to this, block-wise accuracy is presented for two block lengths: 1 (denoted by A1) and 30 s (denoted by A30).

6.2.1 Event detection with the ground-truth context

The system is evaluated first using global acoustic models and then context-dependent acoustic models. At the same time also count-based event priors are evaluated. Event detection results using given ground-truth context labels are presented in Table 3.

The context-dependent acoustic models provide better fitting modeling and this is shown by the consistent increase in the results. Using the count-based event priors increases the system performance in the event detection for most of the contexts in both metrics. The overall accuracy increases from 34.7 to 41.1 while the time resolution error decreases from 86.9 to 83.4. The performance increase is reflected in the block-wise metric with an increase from 10.9 to 14.8 in 1-s block accuracy and 27.0 to 31.2 for 30-s block accuracy.

6.2.2 Event detection with recognized context

The true performance of the system is evaluated using the two steps: context recognition is performed on the test recording and then a set of event models and event priors are chosen according to the recognized context. Event detection results using the proposed two-step system are presented in Table 4. For comparison, the results of a context-independent baseline system [27] is also presented.

The results of the two-step system are slightly lower than the ones presented in Table 3 with the ground-truth context label. This is due to the 9% error in the context recognition step. A wrongly recognized context will lead to choosing the wrong model set and event priors. Even so, the different contexts do contain some common events and some of those events are correctly detected.

Table 3 Monophonic event detection performance based on ground-truth context

	ACC	ER	A1	A30
Global acoustic models				
Uniform event priors	32.3	85.2	10.0	21.9
Count-based event priors	36.6	84.7	12.0	25.8
Context-dependent acoustic models				
Uniform event priors	34.7	86.9	10.9	27.0
Count-based event priors	41.1	83.4	14.8	30.2

Table 4 Monophonic event detection performance comparison with context-independent baseline system and context-dependent system using context recognition

	ACC	ER	A1	A30
Context-independent detection				
No priors, baseline system	28.3	87.0	8.4	17.8
Context-dependent detection				
Uniform event priors	33.8	87.8	10.9	27.0
Count-based event priors	40.1	84.2	14.6	29.8

6.3 Polyphonic event detection

Overlapping events are detected using consecutive passes of the Viterbi algorithm as explained in Section 3.3.2. The average polyphony of the recorded material was estimated based on the annotations, and based on this the number of Viterbi passes was fixed to four.

The system is evaluated first with the ground-truth context label to get the maximum attainable performance of the polyphonic event detection. Later the full system having the context recognition stage is evaluated in order to get the realistic performance evaluation. As discussed in Section 5.2, the CLEAR evaluation metrics are not sensible to be used for polyphonic system output, and only block-wise accuracies are presented. Results for overlapping event detection with ground-truth context labels and recognized context labels are presented in Table 5.

6.3.1 Event detection with the ground-truth context

The consecutive passes of the Viterbi algorithm increase the event detection performance especially when measured on 1-s block-level. On longer 30 s block-level the performance difference is smaller between monophonic output and polyphonic output. The monophonic output can capture small segments of the overlapping events as they become more prominent than other events within

Table 5 Polyphonic event detection results and comparison with monophonic event detection system performance

	Ground-truth context		Recognized context	
	A1	A30	A1	A30
Monophonic system output				
Uniform event priors	10.9	27.0	10.9	27.0
Count-based event priors	14.8	30.2	14.6	29.8
Polyphonic system output				
Uniform event priors	19.8	28.9	18.9	28.2
Count-based event priors	20.4	30.0	19.5	29.4

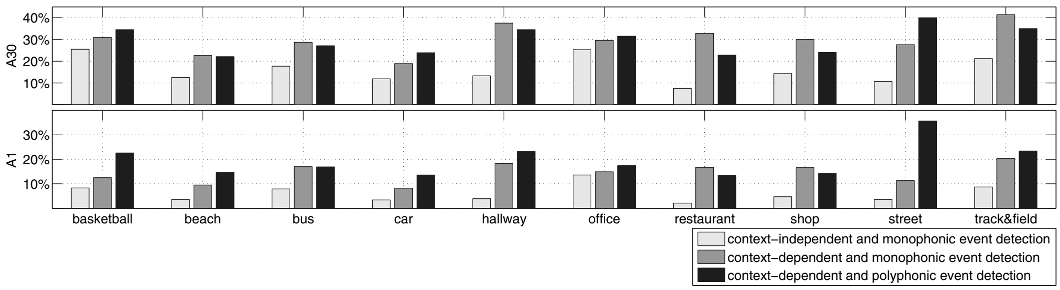


Figure 9 Context-wise results for context-independent baseline system and proposed context-dependent sound event detection systems.

the block. This way the monophonic system can detect many of the overlapping sound events on longer blocks.

6.3.2 Event detection with recognized context

The true performance of the system is evaluated using the context recognizer to get the context label for the test recording. The differences in the performance between the monophonic and polyphonic detection are quite similar to the detection where the true context was given. A slight overall performance decrease is due to the contexts which are not recognized 100% correctly (see Table 2).

6.4 Discussion

The context-dependent sound event detection substantially improves the performance compared to the context-independent detection approach. The improvement is partly due to the context-dependent event selection, and partly due to more accurate sound event modeling within the context. The event selection simplifies the detection task by reducing the number of sound events involved in the process. A context-dependent acoustic model represents particular characteristics of the sound event specific to the context, and provides more accurate results. The two-step classification scheme allows the proposed system to be extended easily with additional contexts later. The training process has to be applied only for the new context to get the context model for the context classification and to get the sound event models for the event detection.

Analysis of the individual contexts reveals interesting performance differences between contexts. Selected context-wise results are presented in Figure 9. Results are presented for three different system configurations: the context-independent baseline system, context-dependent monophonic event detection system using count-based event priors, and context-dependent polyphonic event detection system using count-based event priors. The context-dependent sound event detection approach increases the accuracy on all the studied contexts, especially on the rather complex contexts like street and restaurant. On the other hand, some contexts, like basketball, beach, and office, do not benefit as much.

The proposed overlapping event detection approach provides equal or better performance than prominent event detection approach for most of the contexts. The multiple Viterbi passes increases the detection accuracy in the shorter 1-s blocks relatively more than in 30-s blocks. This property can be exploited when a more responsive detection is required. An impressive improvement of 23% units is achieved in the 1-s block-wise accuracy for the street context, which is probably the noisiest context. On the other hand, the contexts also having a complex auditory scene, the restaurant, and the shop have a slight decrease in the accuracy. Varying complexity per context, i.e., having a different amount of overlapping events present at different times, may require also a different amount of Viterbi passes to overcome this. Examples of the audio recordings used in the evaluations along with their manual annotations and automatically detected sound events are available at arg.cs.tut.fi/demo/CASAbrowser.

7 Conclusion

The benefits of using the context-dependent information in the sound event detection were studied in this article. The proposed approach utilizing the context information comprised a context recognition stage and a sound event detection stage using the information of the recognized context. The evaluation results show that the knowledge of context can be used to substantially increase the acoustic event detection accuracy compared to the context-independent baseline approach. The context information is incorporated in multiple ways into the system. The detection task is simplified by using context-dependent event selection and the acoustic models of the sound events are made more accurate within each context by using context-dependent acoustic modeling. The context-dependent event priors are used to model event probabilities within the context. For example, the detection accuracy in the block-metrics is almost doubled compared to the baseline system. Furthermore, the proposed approach for detecting overlapping sound events increases the responsiveness of the sound event detection by providing better detection accuracy on the shorter 1-s blocks.

Auditory scenes are naturally complex, having usually many overlapping sound events active at the same time. Hence, the detection of overlapping sound events is an important aspect for more robust and realistic sound event detection system. Recent developments in the sound source separation provide interesting possibilities to tackle this problem. In the early studies, sound source separation has already proven to substantially increase the accuracy of the event detection [40]. Further, the event priors for the overlapping sound events are difficult to model because of high number of possible combinations and transitions between them. Latent semantic analysis has emerged as a interesting solution to learn associations between overlapping events [29], but the area requires more studying to apply it efficiently to the overlapping event detection.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Department of Signal Processing, Tampere University of Technology, P.O. Box 553, Tampere, Finland. [2]Nokia Research Center, Visiokatu 3, Tampere, Finland.

Received: 26 June 2012 Accepted: 4 December 2012
Published: 9 January 2013

References

1. AK Dey, Understanding and using context. Person. Ubiquit Comput. **5**, 4–7 (2001)
2. D Wang, GJ Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. (Wiley-IEEE Press, New York, 2006)
3. R Cai, L Lu, A Hanjalic, H Zhang, LH Cai, A flexible framework for key audio effects detection and auditory context inference. IEEE Trans. Audio Speech Lang. Process. **14**(3), 1026–1039 (2006)
4. M Xu, C Xu, L Duan, JS Jin, S Luo, Audio keywords generation for sports video analysis. ACM Trans. Multimed. Comput. Commun. Appl. **4**(2), 1–23 (2008)
5. Y Peng, C Lin, M Sun, K Tsai, in *IEEE International Conference on Multimedia and Expo, 2009. ICME 2009*. Healthcare audio event classification using hidden Markov models and hierarchical hidden Markov models (IEEE Computer Society, New York, NY, USA, 2009), pp. 1218–1221
6. A Härmä, MF McKinney, J Skowronek, in *IEEE International Conference on Multimedia and Expo*. Automatic surveillance of the acoustic activity in our living environment (IEEE Computer Society, Amsterdam Netherlands, 2005), pp. 634–637
7. S Ntalampiras, I Potamitis, N Fakotakis, in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '09*. On acoustic surveillance of hazardous situations (IEEE Computer Society, Washington, DC, USA, 2009), pp. 165–168
8. S Chu, S Narayanan, CCJ Kuo, Environmental sound recognition with time-frequency audio features. IEEE Trans. Audio Speech Lang. Process. **17**(6), 1142–1158 (2009)
9. T Heittola, A Mesaros, A Eronen, T Virtanen, in *18th European Signal Processing Conference*. Audio context recognition using audio event histograms (Aalborg, Denmark, 2010), pp. 1272–1276
10. M Shah, B Mears, C Chakrabarti, A Spanias, in *2012 IEEE International Conference on Emerging Signal Processing Applications (ESPA)*. Lifelogging: archival and retrieval of continuously recorded audio using wearable devices (IEEE Computer Society, Las Vegas, NV, USA, 2012), pp. 99–102
11. G Wichern, J Xue, H Thornburg, B Mechtley, A Spanias, Segmentation, indexing, and retrieval for environmental and natural sounds. IEEE Trans. Audio Speech Lang. Process. **18**(3), 688–707 (2010)
12. AS Bregman, *Auditory Scene Analysis*. (MIT Press, Cambridge MA, 1990)
13. M Bar, The proactive brain: using analogies and associations to generate predictions. Trends Cogn. Sci. **11**(7), 280–289 (2007)
14. A Oliva, A Torralba, The role of context in object recognition. Trends Cogn. Sci. **11**(12), 520–527 (2007)
15. M Niessen, L van Maanen, T Andringa, in *IEEE International Conference on Semantic Computing*. Disambiguating sounds through context (IEEE Computer Society, Santa Clara, CA, USA, 2008), pp. 88–95
16. C Clavel, T Ehrette, G Richard, in *IEEE International Conference on Multimedia and Expo*. Events detection for an audio-based surveillance system (IEEE Computer Society, Los Alamitos, CA, USA, 2005), pp. 1306–1309
17. H Wu, J Mendel, Classification of battlefield ground vehicles using acoustic features and fuzzy logic rule-based classifiers. IEEE Trans. Fuzzy Syst. **15**, 56–72 (2007)
18. L Atlas, G Bernard, S Narayanan, Applications of time-frequency analysis to signals from manufacturing and machine monitoring sensors. Proc. IEEE. **84**(9), 1319–1329 (1996)
19. S Fagerlund, Bird species recognition using support vector machines. EURASIP J. Appl. Signal Process. **2007**, 64–64 (2007)
20. F Kraft, R Malkin, T Schaaf, A Waibel, in *Proceedings of Interspeech*. Temporal ICA for classification of acoustic events in a kitchen environment (International Speech Communication Association, Lisboa, Portugal, 2005), pp. 2689–2692
21. J Chen, AH Kam, J Zhang, N Liu, L Shue, in *Pervasive Computing*. Bathroom activity monitoring based on sound (Springer, Berlin, 2005), pp. 47–61
22. A Temko, C Nadeu, Classification of acoustic events using SVM-based clustering schemes. Pattern Recognit. **39**(4), 682–694 (2006)
23. TH Dat, H Li, in *IEEE International Conference on Acoustics, Speech and Signal Processing*. Probabilistic distance SVM with Hellinger-exponential kernel for sound event classification (IEEE Computer Society, Prague, Czech Republic, 2011), pp. 2272–2275
24. R Stiefelhagen, R Bowers, J(eds) Fiscus, *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. (Springer, Berlin Germany, 2008)
25. X Zhou, X Zhuang, M Liu, H Tang, M Hasegawa-Johnson, T Huang, in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*. HMM-based acoustic event detection with AdaBoost feature selection (Springer, Berlin, Germany, 2008), pp. 345–353
26. X Zhuang, X Zhou, MA Hasegawa-Johnson, TS Huang, Real-world acoustic event detection. Pattern Recognit. Lett. (Pattern Recognition of Non-Speech Audio). **31**(12), 1543–1551 (2010)
27. A Mesaros, T Heittola, A Eronen, T Virtanen, in *18th European Signal Processing Conference*. Acoustic event detection in real-life recordings (Aalborg, Denmark, 2010), pp. 1267–1271
28. M Akbacak, JHL Hansen, Environmental sniffing: noise knowledge estimation for robust speech systems. IEEE Trans. Audio Speech Lang. Process. **15**(2), 465–477 (2007)
29. A Mesaros, H Heittola, A Klapuri, in *19th European Signal Processing Conference*. Latent semantic analysis in sound event detection (Barcelona, Spain, 2011), pp. 1307-1311
30. A Eronen, V Peltonen, J Tuomi, A Klapuri, S Fagerlund, T Sorsa, G Lorho, J Huopaniemi, Audio-based context recognition. IEEE Trans. Audio Speech Lang. Process. **14**, 321–329 (2006)
31. JJ Aucouturier, B Defréville, F Pacher, 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)
32. R Cai, L Lu, A Hanjalic, Co-clustering for auditory scene categorization. IEEE Trans. Multimed. **10**(4), 596–606 (2008)
33. L Lie, A Hanjalic, Text-like segmentation of general audio for content-based retrieval. IEEE Trans. Multimed. **11**(4), 658–669 (2009)
34. LR Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE. **77**(2), 257–286 (1989)

35. D Reynolds, R Rose, Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Trans. Speech Audio Process. **3**, 72–83 (1995)
36. GD Forney, The Viterbi algorithm. Proc. IEEE. **61**(3), 268–278 (1973)
37. M Ryynänen, A Klapuri, in *Proceedings of the 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. Polyphonic music transcription using note event modeling (IEEE Computer Society, New York, NY, USA, 2005), pp. 319–322
38. A Temko, C Nadeu, D Macho, R Malkin, C Zieger, M Omologo, in *Computers in the Human Interaction Loop*, ed. by AH Waibel, R Stiefelhagen. Acoustic event detection and classification (Springer, New York, 2009), pp. 61–73
39. M Grootel, T Andringa, J Krijnders, in *Proceedings of the NAG/DAGA Meeting 2009*. DARES-G1: database of annotated real-world everyday sounds (Rotterdam, Netherlands, 2009), pp. 996–999
40. T Heittola, A Mesaros, T Virtanen, A Eronen, in *Workshop on Machine Listening in Multisource Environments, CHiME2011*. Sound event detection in multisource environments using source separation (Florence, Italy, 2011), pp. 36–40

doi:10.1186/1687-4722-2013-1
Cite this article as: Heittola *et al.*: Context-dependent sound event detection. *EURASIP Journal on Audio, Speech, and Music Processing* 2013 **2013**:1.

PUBLICATION P3

T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound Event Detection in Multisource Environments Using Source Separation," in *Workshop on Machine Listening in Multisource Environments*, (Florence, Italy), pp. 36–40, 2011.

Sound Event Detection in Multisource Environments Using Source Separation

Toni Heittola[1], *Annamaria Mesaros*[1], *Tuomas Virtanen*[1], *Antti Eronen*[2]

[1]Department of Signal Processing, Tampere University of Technology, Tampere, Finland
[2]Nokia Research Center, Tampere, Finland

{toni.heittola, annamaria.mesaros, tuomas.virtanen}@tut.fi, antti.eronen@nokia.com

Abstract

This paper proposes a sound event detection system for natural multisource environments, using a sound source separation front-end. The recognizer aims at detecting sound events from various everyday contexts. The audio is preprocessed using non-negative matrix factorization and separated into four individual signals. Each sound event class is represented by a Hidden Markov Model trained using mel frequency cepstral coefficients extracted from the audio. Each separated signal is used individually for feature extraction and then segmentation and classification of sound events using the Viterbi algorithm. The separation allows detection of a maximum of four overlapping events. The proposed system shows a significant increase in event detection accuracy compared to a system able to output a single sequence of events.

Index Terms: sound event detection, sound source separation, non-negative matrix factorization

1. Introduction

Humans live in a complex audio environment, and have very good skills of following a specific sound source while ignoring or simply acknowledging the others. For example we can follow a conversation in a busy background consisting of other people talking or music. The performance of automatic methods in computational auditory scene analysis (CASA) is much more limited in this task. Acoustic mixture signals contain multiple simultaneously occurring sound events, and machine listening systems are still far from the level of human performance in recognizing them.

Individual sound events can be used to describe an audio scene: they could represent in a symbolic way a scene on a busy street, with cars passing by, car horns and footsteps of people rushing. The different level descriptors represent context (street) and characteristic events (car, car horn, footsteps). Sound event detection and classification aims at processing the acoustic signal and converting it into such symbolic descriptions of the corresponding sound events present at the scene, for applications related to automatic tagging, automatic sound analysis or audio segmentation.

Previous studies related to sound event detection consider audio scenes with overlapping events that are explicitly annotated, but the detection results are presented as a sequence that is assumed to contain only the most prominent event at each time. In this respect, the systems are finding one event at each time, and the evaluation considers the output correct if the detected event is inlcuded in the annotations. The performance of such systems is very limited in case of rich multisource environments.

Sound source separation methods have emerged in recent years for extracting a specific sound source from the mixture. Supervised sound source separation methods are used to separate the signal belonging to one sound source of interest. Unsupervised methods do not use any knowledge about the sound sources, and will usually not separate a specific sound source but a signal with roughly homogeneous spectral content that differs significantly from the background.

In this paper, we propose a sound event detection system that can recognize and temporally locate overlapping sound events in recordings belonging to various audio contexts. The signals are preprocessed using an unsupervised non-negative matrix factorization (NMF) based algorithm in order to separate sound sources into *tracks*. Each of these tracks represents a combination of the physical sources present in the original signal. Event detection is performed on each track. The separation offers the possibility of surpassing the performance levels of previous systems, by giving the possibility of detecting simultaneous events in the multisource environment. The system is evaluated with a database of audio recordings from ten everyday contexts.

The rest of this paper is organized as follows. Section 2 presents a review of related work in event detection and sound source separation. Section 3 presents the sound source separation and Section 4 presents the event detection system. Section 5 explains the database used in the evaluations and the experimental results. Section 6 provides conclusions and suggestions for further study.

2. Related work

Applications of sound event detection from audio include analysis of video sound tracks [1, 2], audio scene recognition [3, 4, 5], audio context recognition [6] or plain acoustic event detection [7]. The cited studies are done on small sets of sound events and small set of environments, and usually the sound events and audio examples are chosen so to minimize overlapping between different categories. In case of overlapping sound events, the annotation considers the most prominent one. There are few studies that consider the case of overlapping sound events. In [7] and [8], the annotation was done to include overlapping events, but the output of the systems is a sequence of non-overlapping events. The detection result ideally consists of a sequence of the most prominent sound events, and the evaluation metric in [7] is developed for that situation. To our knowledge, there is no work that considers modeling and detecting overlapping events for event detection.

The system we presented in [8] for event detection in real life recordings, is based on hidden Markov models (HMM). We trained HMMs for 61 sound event classes using mel-frequency cepstral coefficients (MFCC) extracted from the acoustic mix-

[1]This work was financially supported by the Academy of Finland.

ture signal. For event detection, the Viterbi algorithm was used to decode the best path through the HMM states, with the 61 model HMMs connected into a network. The system was evaluated against annotations that mark overlapping events, and the detection accuracy is therefore limited by the possibility of decoding only one event at each time, while the annotations contained simultaneous events. The sound separation will help overcome this limitation.

Sound source separation aims at separating a mixture signal consisting of multiple additive sources into source signals. Recently, NMF based source separation has produced good results in many applications [9]. The basic NMF separates a signal into sources in an unsupervised manner, i.e., without prior knowledge about the sources.

Supervised source separation utilizes some prior information about the sources. The prior information can include detailed models for the spectra of the source of interest [10, 11, 12], or pitch of the sound obtained by a pitch estimation algorithm in a preprocessing stage [13, 14]. These algorithms have been used to separate mixture signals and to classify the resulting sources in speech [10, 12], singing [13], instrument recognition [14], or music transcription applications [11].

3. Sound source separation

In sound source separation, a given input audio signal which consists of multiple overlapping sounds (mixture signal) is decomposed into its sound sources (ideally). For our sound event detection, we will use a sound source separation method that is based on non-negative matrix factorization of the magnitude spectrogram of the mixture signal [9].

When applied on a spectrogram representation of audio, NMF models the signal as a sum of components, each of which has a fixed magnitude spectrum and a time-varying gain. Since the algorithm is unsupervised, we cannot strictly control the outcome of the factorization, but the components correspond to redundant sound objects in the mixture signal. Each sound source in the mixture signal can become represented as the sum of one or more components. Each component can contain parts from one or more sound sources, but typically the factorization achieves good separation of sound sources.

The processing steps of our NMF based separation algorithm are as follows:

1. Window the input signal into frames using a 60 ms Hamming windows with a 25 % overlap and calculate the complex-valued spectrum $X_t(f)$ in each frame t using the fast Fourier transform. Here f denotes the discrete frequency index. Absolute values of the spectra are stored in to a magnitude spectrogram matrix $[\mathbf{X}]_{f,t} = |X_t(f)|$.
2. Calculate the non-negative matrix factorization $\mathbf{X} \approx \mathbf{SA}$ by minimizing the Kullback-Leibler divergence between the original spectrogram and a reconstructed spectrogram [15]. The number of components is fixed to four in our system. The initial reconstruction of the magnitude spectrogram matrix $\mathbf{Y}^n$ of component n is obtained as $[\mathbf{Y}^n]_{f,t} = [\mathbf{S}]_{f,n}[\mathbf{A}]_{n,t}$.
3. Reconstruct the complex spectrum $Y_t(f)^n$ of component n in frame t and frequency f as $Y_t(f)^n = X_t(f)[\mathbf{Y}^n]_{f,t}/(\sum_m[\mathbf{Y}^m]_{f,t})$. This corresponds to Wiener filtering where the source power spectrum estimate is given by the initial reconstruction of the magnitude spectrogram of the component, and the noise power spectrum estimate is given by the sum of the other components.
4. Convert the complex spectrum of each component in each frame to time domain by inverse fast Fourier transform, and combine the frames using overlap-add. The resulting time-domain signal of an individual component is dubbed a *track*.

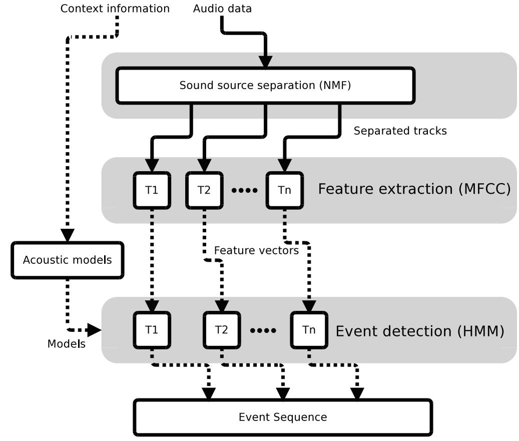


Figure 1: *System overview.*

It is clear that environmental sounds with diverse characteristics cannot be accurately modeled with the NMF model, i.e., with a fixed spectrum and time-varying gain. However, the reconstruction of tracks by Wiener filtering explains better the functioning of the algorithm: the time-varying Wiener filter of each component separates a track which contains roughly homogeneous spectral content that differs significantly from the other tracks.

The resulting separate tracks do not represent exact physical sound sources (like one track would be only sounds of footsteps), but a combination on the physical sources present in the signal. Sound event detection will be performed on each of the separated tracks. In this paper we are splitting the original multisource spectrum into four tracks. This limits the sound event detection to finding a maximum of four simultaneous sound events, in agreement with the average polyphony of our database.

4. Sound event detection

The overall system scheme is presented in Figure 1. Sound source separation is applied on the mixture signal to produce the separated tracks $(T1, T2, ..., Tn)$. Feature extraction and event detection is performed on each of these tracks separately. The results from different tracks are collected and combined into a multisource symbolic representation of the original signal, based on the total number of sound events that are recognized. This representation is then evaluated against ground truth annotations.

The event detection system consists of event class models trained from real-world audio recordings. Each event model is represented by a three-state left-to-right HMM with 16 Gaussians per state. The set of features used for constructing the models are the MFCCs (static, delta and acceleration): 16 MFCCs calculated on 20 ms windows with 50 % overlap.

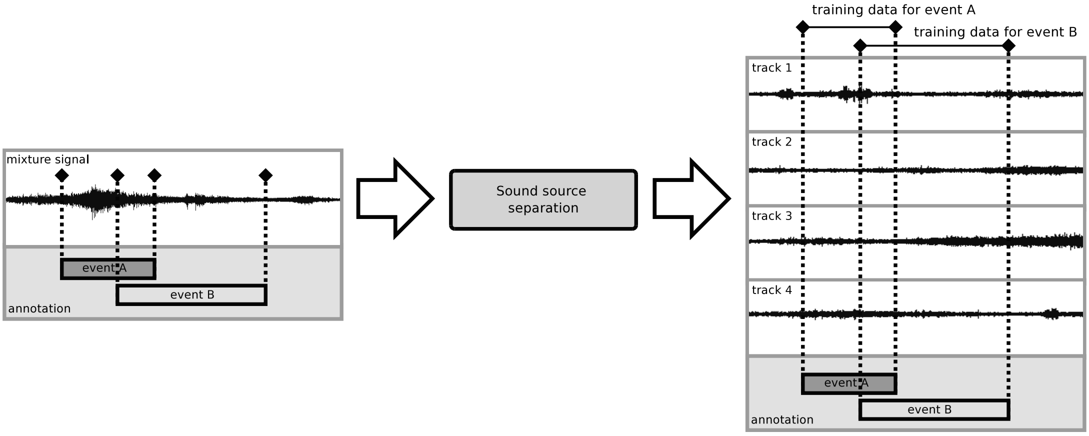


Figure 2: *Separation and segmentation procedure of the original audio for training sound event models.*

4.1. Model training

Each event instance annotated in the database represents one example for the event models. Regions of the sound that contain overlapping events were assigned to the relevant event classes and will be used as examples for all overlapping classes.

The training data is preprocessed using the described NMF-based method. Each recording is separated into four tracks and the annotations are used to provide the segmentation into event instances. Because of the unsupervised sound separation, we do not have any knowledge about which track contains what event from annotations. Therefore we assign the annotated event segment in all the separated tracks to training the annotated class. The assumption behind this is that in training, the tracks that do not contain relevant event classes will average out and the models will learn a reliable representation of the sound events. In general, the model should learn the acoustic representation of the event and average out the extra information. Figure 2 illustrates the procedure of assigning segments for training the sound event models.

We include a universal background model (UBM). This model represents the overall properties of the data and is trained by pooling all training material together. Its role in the event detection system is to capture regions when no events of interest are detected. This may happen at silent spots in the audio or in cases where the models do not score high enough to be considered plausible by the decoder.

4.2. Event Detection

For event detection, the event models are connected into a network HMM, having equal transition probabilities from one event model to another. The output of the detection step is an unrestricted sequence of the most likely model labels: any sound event can follow any other and there is no limit for the number of events. An optimal sequence of events is decoded using the Viterbi algorithm. The event detection is performed for each separated track. Then, the results from individual tracks are combined into a single description of the original audio. The final output contains timestamps and labels for all the recognized events; overlapping events from two or more tracks that carry the same label are combined into one compact representation.

5. Evaluation

The sound event detection system is trained and tested using an audio database collected from real-life contexts. The sound source separation based method is compared with a baseline system which is trained and tested on mixture signals. A detailed description of the approach used for constructing the baseline system can be found in [8]. The training and testing are done in a context-dependent manner, meaning that the number of sound event models trained and connected into a network for detection is limited to the events that are found in the annotation of the considered audio context.

5.1. Database

The material for the database was gathered by recording 10 to 30 minute long audio in ten different contexts. The selected audio contexts were basketball game, beach, inside a bus, inside a car, hallway, office, restaurant, grocery shop, street and stadium with track and field sports. The audio was recorded using binaural microphones placed inside the ears of the person recording. Each context is represented by 8 to 14 recordings, to a total of 103 recordings included in the database. In this study we are using monophonic versions of the recordings, i.e., the two channels are averaged to one channel.

The recordings were manually annotated indicating the start and end time of all clearly audible sound events in the auditory scene. Annotated sound events present in the recordings were grouped into 61 event classes. The event classes include e.g. speech, laughter, applause, car door, road, dishes, door, chair, music, and footsteps. Each context contains 9 to 16 event classes and many event classes appear in multiple contexts. There are also event classes which are context specific. The context-specific training and testing limits the number of models to 9-16 per context instead of training all 61 classes as we did in our previous work, and the material used for the training is also gathered only from the specific context.

5.2. Metric

Most of the previous studies found in the literature are concentrated on detecting non-overlapping events and the metrics presented in them are best suited for evaluating the monophonic output of the detection system. In the CLEAR evaluation [7], two metrics were defined for the sound event detection, one for detection accuracy and one for the temporal resolution of the detection. The detection accuracy was defined as the F-score between precision and recall. A detected event was regarded as correct if there was a certain degree of overlapping with an event with the same label in the annotation. The temporal resolution error was calculated by counting the entire amount of time that was wrongly attributed to events, divided by the total amount time covered by the events. The exact description of the two metrics can be found in [7]. We consider that these metrics are hard to interpret for evaluating an output with overlapping events, as it will be shown further in an example.

The recall of the system is limited by the number of events it can output, compared to the number of events that are annotated. As a consequence, even if the output contains only correct events, the accuracy is limited. The temporal resolution error represents all the erroneously attributed time, including events wrongly recognized and events missed altogether by the lack of sufficient polyphony in the detection. They are therefore complementary and tied to the polyphony of the annotation. This complicates optimization of the detection system into finding a balance between the two.

In order to tackle this problem and give a single understandable metric, we propose a block-wise accuracy for polyphonic case. This metric will evaluate how well the events detected in non-overlapping time blocks coincide with the annotations. The detected events are regarded only at the block level, for example within 30 seconds. This metric is designed for applications requiring fairly coarse time resolution, placing more importance into finding the right events within the block than finding their exact location.

Inside the blocks, we calculate precision and recall. Precision is defined as the number of correctly detected sound event classes divided by the total number of event classes detected within the block. Recall is defined as the number of correctly detected sound event classes divided by the number of all reference event classes within the block. We calculate the accuracy in each block by the F-score, based on precision and recall by the formula:

$$fscore = \frac{2 * precision * recall}{precision + recall} \tag{1}$$

An illustration of how this metric works can be seen in Figure 3. In block 1, the reference events are A, B, C and D; the system output contains A, C and D. The A and C are correctly detected. This means that for this block, precision is 2 out of 3 (2/3) and recall 2 is out of 4 (2/4). The calculated accuracy for this block is 57.1 %. For the entire example, the average block accuracy is 57.3 %.

For comparison, the CLEAR metrics calculated on the same illustration are the following: precision is 6/10 and recall is 6/9, resulting detection accuracy of 63.2 %. For calculating the time resolution error we count the units that are wrongly labeled/missed: there are 56 of them. The total number of units covered by the annotated events is 51, with a resulting time resolution error of 109.8 %.

5.3. Results

The database is divided into sets in a five-fold manner. One set is used as development set and the remaining sets are used for evaluating the system. Inside a set, 70 % of the material is used for training and 30 % for the testing.

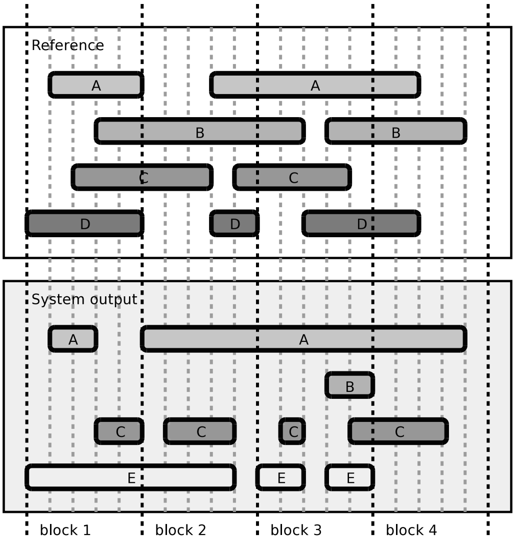


Figure 3: *Block-wise accuracy for sound event detection.*

The non-restricted Viterbi search for the optimum path results in an output containing a very large number of short events, up to ten times more events than the total annotated events. To control the length of the events, we introduce an extra cost at the inter-model transitions. This will result in the Viterbi path staying longer in each model if the cost of transitioning to a new model is higher. The development set is used to search the optimum value for this parameter. The cost value was chosen to be the one that produced a reasonable number of output events - order-wise close to the number of reference (ground truth) events.

The baseline system is trained and tested using the original mixture signal; the audio segment examples for training the classes are extracted based on the annotations, and the same region of audio was included in all annotated overlapping classes too. The baseline system uses the same development set as the proposed system, but with independent cost parameter search.

The event detection results for the baseline system and the proposed system are presented in Table 1. The overall performance of the baseline system is 25.8 % for evaluating precision and recall within 30 second blocks. This value is lower than the accuracy presented in [8], where the system was using more general models and outputting only a sequence of events. As presented in Section 5.2, the proposed block-wise accuracy is lower than the CLEAR evaluation accuracy. The 30 % performance calculated according to the CLEAR evaluation metrics is therefore meaningless without mentioning at the same time the time resolution error, which was 84 %. The block-wise accuracy could be seen as the system performance in detecting the correct events with a coarse time resolution, representing in a way a combination of the CLEAR accuracy and time resolution performance (opposite of the time resolution error).

Overall performance of the detection increases significantly by using sound source separation as preprocessing in training of the models and also in testing. Context-wise, the proposed system performs much better than the baseline system, almost doubling the overall accuracy. Individual contexts show 17 to 38 percent units improvement.

Table 1: *Sound event detection results, accuracy calculated using the block-wise accuracy metric inside 30 second blocks.*

	baseline system	proposed system
Overall	28.2	52.6
Context		
Basketball	30.3	68.2
Beach	23.0	38.7
Bus	24.4	57.6
Car	18.8	46.7
Hallway	37.0	51.1
Office	30.1	49.7
Restaurant	25.4	54.2
Shop	27.7	56.2
Street	26.4	50.1
Track & Field	41.7	57.4

The sound source separation algorithm brings important improvement not just in the numbers, but conceptually. The proposed system is able to detect overlapping events, whereas the baseline system is only producing monophonic output.

6. Conclusions

This paper presented a sound event detection system capable of detecting overlapping events in natural multisource environments. The audio is preprocessed in the sound source separation stage, and separated into four individual tracks representing combinations of the physical sources present in the signal. Sound event detection is applied to each track separately. We use recordings from ten everyday environments. In the evaluations, sound source separation was found to substantially increase the sound detection accuracy. In addition to this, the proposed system produces a conceptually accurate symbolic representation of the environment by detecting overlapping events. Thus, we conclude that the proposed method improves the sound event detection performance by producing more accurate and more realistic results.

7. References

[1] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L.-H. Cai, “A flexible framework for key audio effects detection and auditory context inference,” *IEEE Trans. on Audio, Speech and Language Process.*, vol. 14, no. 3, pp. 1026–1039, 2006.

[2] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, “Audio keywords generation for sports video analysis,” *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 4, no. 2, pp. 1–23, 2008.

[3] C. Clavel, T. Ehrette, and G. Richard, “Events detection for an audio-based surveillance system,” in *Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, ICME 2005, July 6-9, 2005, Amsterdam, The Netherlands*, pp. 1306–1309, 2005.

[4] A. Härmä, M. F. McKinney, and J. Skowronek, “Automatic surveillance of the acoustic activity in our living environment,” *IEEE International Conference on Multimedia and Expo*, pp. 634–637, 2005.

[5] A. Eronen, V. Peltonen, J. Tuomi, A. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, “Audio-based context recognition,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 14, pp. 321–329, Jan. 2006.

[6] S. Chu, S. Narayanan, and C.-C. J. Kuo, “Environmental sound recognition with time-frequency audio features,” *IEEE Trans. on Audio, Speech and Language Process.*, vol. 17, no. 6, pp. 1142–1158, 2009.

[7] R. Stiefelhagen, R. Bowers, and J. Fiscus, eds., *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007.* Berlin, Heidelberg: Springer-Verlag, 2008.

[8] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, “Acoustic event detection in real-life recordings,” in *18th European Signal Processing Conference*, (Aalborg, Denmark), pp. 1267–1271, 2010.

[9] T. Virtanen, “Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 15, pp. 1066–1074, March 2007.

[10] B. Raj, R. Singh, P. Smaragdis, and B. R. R. Singh, “Recognizing speech from simultaneous speakers,” in *Proc. Interspeech*, pp. 3317–3320, 2005.

[11] J. Paulus and T. Virtanen, “Drum transcription with nonnegative spectrogram factorisation,” in *EUSIPCO*, pp. 4–8, 2005.

[12] P. Smaragdis, “Convolutive speech bases and their application to supervised speech separation,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, pp. 1 –12, jan. 2007.

[13] A. Mesaros, T. Virtanen, and A. Klapuri, “Singer identification in polyphonic music using vocal separation and pattern recognition methods,” in *Proceedings 7th Internationl Society for Music Information Retrieval Conference (ISMIR)*, (Vienna, Austria), pp. 375–378, 2007.

[14] T. Heittola, A. Klapuri, and T. Virtanen, “Musical instrument recognition in polyphonic audio using source-filter model for sound separation,” in *Proceedings 10th Internationl Society for Music Information Retrieval Conference (ISMIR)*, (Kobe, Japan), pp. 327–332, 2009.

[15] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *NIPS*, pp. 556–562, MIT Press, 2000.

PUBLICATION P4

T. Heittola, A. Mesaros, T. Virtanen, and M. Gabbouj, “Supervised Model Training for Overlapping Sound Events Based on Unsupervised Source Separation,” in *Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing*, (Vancouver, Canada), pp. 8677–8681, 2013.

SUPERVISED MODEL TRAINING FOR OVERLAPPING SOUND EVENTS BASED ON UNSUPERVISED SOURCE SEPARATION

Toni Heittola⋆ *Annamaria Mesaros*† *Tuomas Virtanen*⋆ *Moncef Gabbouj*⋆

⋆ Department of Signal Processing, Tampere University of Technology
† Department of Information and Computer Science, Aalto University

ABSTRACT

Sound event detection is addressed in the presence of overlapping sounds. Unsupervised sound source separation into streams is used as a preprocessing step to minimize the interference of overlapping events. This poses a problem in supervised model training, since there is no knowledge about which separated stream contains the targeted sound source. We propose two iterative approaches based on EM algorithm to select the most likely stream to contain the target sound: one by selecting always the most likely stream and another one by gradually eliminating the most unlikely streams from the training. The approaches were evaluated with a database containing recordings from various contexts, against the baseline system trained without applying stream selection. Both proposed approaches were found to give a reasonable increase of 8 percentage units in the detection accuracy.

Index Terms— acoustic event detection, sound source separation, supervised model training, acoustic pattern recognition

1. INTRODUCTION

A *sound event* is a segment of audio which can be characterized and identified by a textual label. Sound events can be used to describe and understand the human and social activities. Automatic sound event detection aims at processing a continuous acoustic signal and converting it into a sequence of event labels with associated start times and end times. The sound event detection can be utilized in a variety of application areas, including context-based indexing and retrieval in multimedia databases [1, 2], unobtrusive monitoring in health care [3], and audio-based surveillance [4]. Furthermore, the detected events can be used as mid-level-representation in other research areas, e.g. audio context recognition [5, 6], automatic tagging [7], and audio segmentation [8].

Early research on sound event detection concentrated on detecting only one sound event at a time, considerably simplifying the detection problem [9, 10, 11]. Everyday auditory scenes are usually complex in sound events, having multiple overlapping sound events active at the same time. If an algorithm that detects only a single event at a time is applied to material consisting of overlapping events, the majority of detection errors will be caused by temporally overlapping sound events. In order to detect all sound events, a way to deal with overlapping events is needed. Recently, the problem of overlapping events has been addressed at various levels of the detection process. At the signal level, unsupervised sound source separation can be used to minimize the acoustical interference of overlapping sound sources [12]. In the acoustic model training, the overlapping events can be taken into account by modeling all possible event combinations as new intermediate classes [13, 14]. In the event detection stage, overlapping events can be detected with multiple iterative detection passes and by excluding already detected events from the following detection iterations until the desired amount of overlapping events have been reached [15]. In addition to these approaches, multiple audio signals and sound source localization methods along with video based methods can be used to better handle overlapping event in the detection [16].

In this paper, we tackle the problem of overlapping events by applying unsupervised sound source separation as a preprocessing stage for the event detection. In the source separation stage, the mixture signal is split into *streams* containing roughly homogeneous spectral content, each differing significantly from the other streams. Following the concept of noise adaptive training used in robust speech recognition [17], the same signal enhancement method should be applied both before model training and detection stages. Due to the unsupervised nature of the separation, there is no knowledge about which sound source is separated into which stream, making it challenging to take full advantage of the separated audio as such in the supervised model training.

We propose a method to train reliable acoustic event models by iteratively selecting the most appropriate training material from audio separated in an unsupervised manner. Prior knowledge about the temporal location of events given by annotations is used to get initial models for event classes. Two alternative approaches using expectation maximization (EM) algorithm to select the stream that contains the target sound are proposed: one selecting always the most likely stream and another gradually eliminating the most unlikely streams from the training. The proposed method is evaluated with a database recorded in realistic environments with a high degree of overlapping sound events. The method is compared to the baseline system trained without the stream selection. At the general level, this work extends our context-dependent sound event detection system presented in [12] with event priors and proposed model training approach.

The rest of this paper is organized as follows. Section 2 presents the sound event detection system for overlapping events, and Section 3 explains the model training using recordings with overlapping events. Section 4 presents the experimental results, and Section 5 discusses them. Section 6 provides conclusions and future work.

2. SOUND EVENT DETECTION

The overview of the sound event detection system is presented in Figure 1. Sound source separation is applied on the mixture signal to produce the streams $(S1, S2, S3, S4)$. In this study, the number of streams is fixed to four. Feature extraction and sound event

This work was financially supported by Academy of Finland under the grants 258708 and 265024

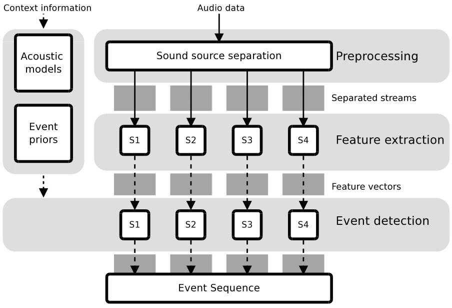


Fig. 1. Overview of sound event detection system.

detection are performed on each of these streams separately and the resulting event sequences are combined into a multi-source symbolic representation of the original signal.

In the event detection stage, a given context is used to select a context-specific set of events with context-specific acoustic event models and prior probabilities. This provides more accurate modeling, since many sound events are acoustically dissimilar across contexts [15]. Furthermore, some sound events are more likely than others, and the differences in occurrence rates are even more obvious between contexts.

2.1. Source separation

In the source separation stage, a given input audio signal that consists of multiple overlapping sounds (mixture signal) is decomposed into its sound sources (ideally). The proposed system utilizes an unsupervised sound source separation method based on non-negative matrix factorization (NMF) of the magnitude spectrogram of the mixture signal [18]. The method models the mixture signal as a sum of components, each having a fixed magnitude spectrum and a time-varying gain. Due to the unsupervised nature of the method, the outcome of the factorization cannot be strictly controlled. Sound sources in the mixture signal may get represented as the sum of one or more components, and at the same time each component can contain parts from one or more sound sources. However, typically the factorization achieves good separation of sound sources. A more detailed description of the separation algorithm is presented in [12].

Most of the sound events have diverse characteristics and they cannot be accurately modeled with fixed spectrums and time-varying gains. However, the function of the algorithm is better explained by reconstructing the streams with Wiener filtering: a time-varying Wiener filter of each component separates a stream which contains roughly homogeneous spectral content that differs significantly from the other streams. The resulting streams represent a combination of the physical sources present in the mixture signal, rather than exact physical sound sources. In this paper, the original multisource spectrum is split into four streams (number of components) limiting the the event detection to finding a maximum of four simultaneous sound events. This is in agreement with the average amount of overlapping events in our evaluation database.

2.2. Event models

The coarse shape of the power spectrum of the input signal is represented with 16 mel-frequency cepstral coefficients (MFCCs). In order to describe the dynamic properties of the cepstrum, first and second time derivatives of the static coefficients are also utilized. Features are calculated in 20 ms frames with 50 % overlap.

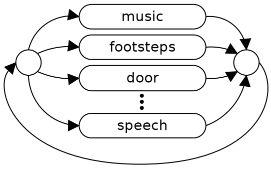


Fig. 2. Fully-connected sound event model network.

Continuous-density hidden Markov models (HMMs) with three state left-to-right topology are used to model sound-event-conditional feature distributions. The probability density functions of observations in each state are modeled with a mixture of multivariate Gaussian density functions (16 Gaussians). The model training process is described in detail in Section 3. In the testing stage, the trained sound event models are connected into a network with transitions from each model to any other. A model network is shown in Figure 2.

Manually annotated training recordings are used to estimate the event priors, i.e., transition probabilities in the network. Annotated events are regarded as a separate entities, and their event-lengths are accumulated (in precision of seconds). Normalized lengths of each event class are used as event priors.

2.3. Detection

Sound event detection is applied separately for each stream. This is obtained by applying Viterbi decoding inside the network of sound event models. The alignment of states and observations given by the Viterbi algorithm produces estimates of event segment boundaries and event labels. Detection results from each stream are merged into a single set of events as in [12].

When calculating the path cost through the model network, the balance between likelihoods provided by the event priors and the acoustic models is adjusted using a weight parameter. The number of events in the resulting event sequence is controlled by using a cost for inter-event transitions. Both these parameters are experimentally chosen using a development set, and are tuned so that the output has approximately equal amount of events as the manually annotated ground truth. A more detailed description of the detection stage is presented in [15].

3. MODEL TRAINING

In the the source separation stage, each original recording is split into four audio streams. The training material for an event class is selected based on annotated time-segments. Since the source separation is done in an unsupervised manner, there is no exact knowledge about which stream contains most suitable training material for the target sound event class. The problem is to select which of the four streams contains the target event class. In this work, we assume that there is always one single stream containing the target sound, and other three streams are regarded to contain overlapping events. The stream selection for training is illustrated in Figure 3.

Regardless of the stream selection, the overlapping events might still cause some interference and variability to the training material. However, this is assumed to be averaged out in the model training

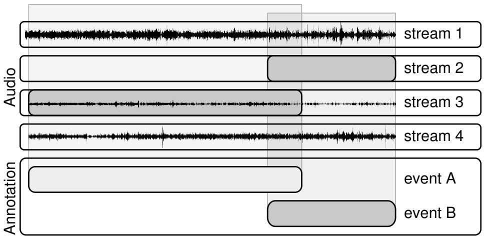


Fig. 3. Separated audio streams and material selection for model training. Annotated events A and B are separated into distinct streams 3 and 2.

due to the large training set, and the models will learn a reliable representation of the target sound events.

3.1. Expectation maximization algorithm

The iterative stream selection is based on expectation-maximization (EM) algorithm [19]. In order to simplify the notation, we present training of a single event class model λ. The described procedure is identical for each of the classes. An audio segment extracted from an annotated time-segment s in a stream with index m is denoted as $x_{s,m}$. Let us denote a set of events that are annotated to content target class by set $\mathcal{C}$.

The EM algorithm is used to iteratively associate subset of the $x_{s,m}$ for training acoustic model λ for a sound class. Acoustic model λ is initialized by training a model using all annotated time-segments S from all four separated streams, $x_{\mathcal{C},1:4}$. Notation $x_{\mathcal{C},1:4}$ denotes all the x indexed by event set $s \in \mathcal{C}$ and $m \in [1,4]$. After this the EM algorithm operates iteratively repeating the E step and M step while the value of the likelihood function $P(\lambda \mid x_{\mathcal{C},1:4})$ is maximized at each iteration. Using Bayesian expansion, expression to be maximized is $P(x_{\mathcal{C},1:4} \mid \lambda)$, which is defined as

$$P(x_{\mathcal{C},1:4} \mid \lambda) \equiv \sum_{s\in\mathcal{C}} \sum_{m} P(x_{s,m}, a_s = m \mid \lambda), \tag{1}$$

where latent variable a_s denotes the index of the stream that contains the target event. This can be further expanded into

$$P(x_{\mathcal{C},1:4} \mid \lambda) = \sum_{s\in\mathcal{C}} \sum_{m} P(x_{s,m} \mid \lambda) P(a_s = m \mid x_{s,m}, \lambda). \tag{2}$$

Above, $P(x_{s,m} \mid \lambda)$ is the likelihood of $x_{s,m}$ for the HMM event model. Let us denote the posterior probability $P(a_s = m \mid x_{s,m}, \lambda)$ by $a_{s,m}$. The EM algorithm iterates over expectation – calculating $a_{s,m}$ and maximization – recalculating model λ:

$$\text{(E):} \qquad a_{s,m} = P(a_s = m \mid x_{s,m}, \lambda) \tag{3}$$

$$\text{(M):} \qquad \lambda \leftarrow \arg\max_{\lambda} \sum_{s\in\mathcal{C}} \sum_{m} P(x_{s,m} \mid \lambda)\, a_{s,m}. \tag{4}$$

The expectation step represents the stream selection, and is given as

$$a_{s,m} = \frac{P(x_{s,m} \mid \lambda)}{\sum_{m'} P(x_{s,m'} \mid \lambda)}. \tag{5}$$

The maximization step in Eq. 4 represents the training of the new models and is solved by conventional Baum-Welch algorithm used to train HMMs.

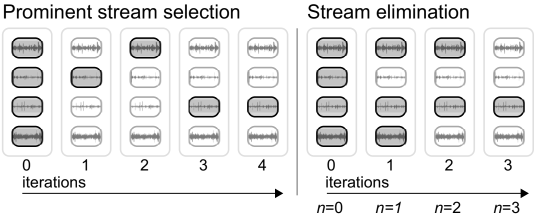


Fig. 4. Example of proposed stream selection approaches. Prominent stream selection: in each iteration only one $a_{s,m}$ is set to one, rest are zero. Stream elimination: in each iteration one more $a_{s,m}$ is set to zero.

In order to simplify the maximization step, a is made binary as described in the next section. As a result of this, only those $x_{s,m}$ for which $a_{s,m} = 1$ are used in the maximization. This avoids using weighted observations so that standard HMM training algorithms can be used.

3.2. Stream selection

We propose two approaches to make a binary. In the first one, only the most likely stream is selected, i.e., $a_{s,m}$ having the highest likelihood among $a_{s,1:4}$ is set to one and $a_{s,m'}$ for other m is set to zero. This approach is later denoted as *prominent stream selection*.

In the second one, the n smallest $a_{s,m}$ among $a_{s,1:4}$ are set to zero, i.e. eliminated. We set n equal to the iteration count. This approach is later denoted as *stream elimination*. The illustration of how the stream selection approaches are applied to one training instance is shown in Figure 4.

Prominent stream selection is repeated until convergence, i.e. the stream indexes do not change. The stream elimination is repeated until only one stream is left.

4. SYSTEM EVALUATION

The sound event detection system is trained and tested using an audio database collected from real-life contexts. The training and testing are done in a context-dependent manner, using context-dependent count-based priors and acoustical models.

4.1. Database

The database consists of 103 recordings ranging from 10 to 30 minutes resulting in total 1133 minutes of audio. The recordings were collected from ten audio contexts: basketball game, beach, inside a bus, inside a car, hallways, inside an office facility, restaurant, grocery shop, street, and stadium with track and field events. There were 8-14 recordings made in each context using binaural microphones placed inside the ears of the person recording. In this study we are using monophonic versions of the recordings, i.e., the two channels are averaged to one channel.

All clearly audible sound events in the recordings were manually annotated by indicating the start and end times of the sound events. Total of 61 distinct event classes are used in the study. The event classes include e.g. speech, laughter, applause, car door, road, dishes, door, chair, music, and footsteps. The number of events that can be active at the same time was not limited. In this sense, the

	A1	pre / rec	A30	pre / rec
Baseline	**36.7±2.4**	33.1 / 41.2	**57.2±2.2**	53.8 / 61.2
Prominent stream selection				
Iteration 1	42.8±5.2	38.9 / 47.6	60.6±3.6	58.1 / 63.4
Iteration 2	43.8±4.4	39.4 / 49.3	60.6±2.3	57.7 / 63.9
Iteration 3	**44.5±5.9**	40.0 / 50.2	**60.9±2.9**	58.1 / 64.1
Iteration 4	44.1±5.8	39.7 / 49.8	60.5±2.3	57.8 / 63.6
Stream elimination				
Iteration 1, *n*=1	37.9±2.3	34.3 / 42.4	58.4±0.7	55.2 / 62.0
Iteration 2, *n*=2	40.4±4.0	36.3 / 45.6	60.2±1.7	57.0 / 63.9
Iteration 3, *n*=3	**44.9±4.7**	40.2 / 51.1	**60.8±2.8**	58.0 / 64.0

Table 1. Sound event detection accuracy, calculated based on precision (pre) and recall (rec), for the baseline and systems using proposed stream selection approaches.

recordings can be regarded as polyphonic. Usually in a natural auditory scene the event classes are not equally represented. While many event classes are very common and shared between multiple contexts (e.g. speech), some event classes can be quite rare or they are highly context-specific (e.g. referee whistle in basketball game or pressure release noise inside the bus). A more detailed description of the database and event class statistics can be found in [11].

4.2. Performance evaluation

For evaluating the system output, we will use the block-wise detection accuracy metric proposed in [12]. This metric evaluates how well the events detected in non-overlapping time blocks coincide with the annotations. The detected events are regarded only at the block level, and in this study we are using two block lengths: one second (denoted by A1) and 30 seconds (denoted by A30).

Inside a block, precision and recall are calculated, and block-wise detection accuracy is represented by the F-score. An event is regarded as correctly detected if it has been detected and annotated somewhere within the considered block.

4.3. Results

The detection accuracy of the models produced by the proposed stream selection approaches was evaluated and compared against a baseline system which is using event models trained without stream selection. The event models used in the baseline are also used as initial models for the stream selection process.

The evaluation database was split randomly into five equal-sized sets, with one set being used as test data and other four for training the system. The split was done five times for a five-fold cross-validation setup. One fold was used in the development stage for determining parameters for the event sequence decoding. The evaluation results are presented as the average of the other four folds.

The event detection results for the baseline system and the proposed stream selection approaches are presented in Table 1 (best performance highlighted). The results show average detection accuracy along with 95 % confidence interval. The number of iteration steps for the prominent stream selection approach was four, since only minimal changes (0.1 % change) were noticed after the fourth iteration. In the stream elimination approach, the elimination parameter *n* was increased with one in each iteration. After three iterations only the most likely stream was left and the iteration was ended.

Detection accuracy increases steadily with both of the selection approaches throughout the iterations. In the end, both approaches provide similar level of increase in the block-wise accuracy compared to the baseline system. In the one second block-level, the improvement in detection accuracy is over 8 percentage units for both selection approaches after three iterations. In the 30 second block-level, the improvement is more modest being only 3 percentage units. The increased accuracy of the detection is especially noticeable in the recall of the detection for both block-levels, i.e. bigger portion of annotated events are correctly detected.

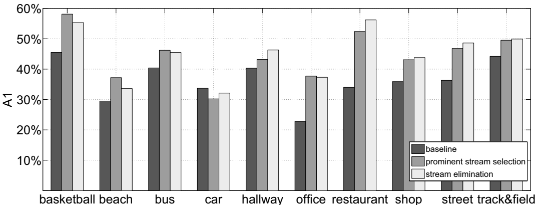


Fig. 5. Context-wise detection accuracy (A1) after three stream selection iterations, along with the baseline accuracy.

The context-wise results are shown in Figure 5. For restaurant and office, the proposed stream selection approach will give significant improvement, whereas for recording made inside a car, the detection accuracy even drops a bit. This may be due to the fact that car environment is very noisy and the degree of overlapping between events is low.

5. DISCUSSIONS

The main difficulty when using the prominent stream selection approach is to know how many iterations are needed. In this study we stopped the number of iterations at four, but in fact the maximum detection accuracy was obtained after three iterations. Results in Table 1 show that accuracy does not change significantly after the first iteration. This means that the first iteration already selects most of the correct streams for each target class.

The stream elimination approach is more straightforward, as one needs to perform iterations until only one stream is left. In this approach, the detection accuracy increases gradually, reaching maximum at the end of the process.

Compared to previous work using sound source separation [12], the presented work increases significantly (52.6 % to 60.9 % in A30) the performance through using event priors and the proposed stream selection method in training the models. Compared to detection on polyphonic audio, that does not use any source separation, the performance is more than doubled [15].

6. CONCLUSIONS

A method for training acoustic event models from acoustic material containing high degree of overlapping events was proposed. In the preprocessing stage, the unsupervised sound source separation was applied to the audio signal in order to minimize the interference of overlapping events. The most appropriate training material for the target sound class was selected iteratively from the separated audio streams using an EM algorithm. The approaches for selecting streams work by selecting the most likely or eliminating the most unlikely streams. Both approaches were found to give reasonable increase in the detection accuracy compared to the baseline system. This highlights the benefits of carefully selecting training material.

7. REFERENCES

[1] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L-H. Cai, “A flexible framework for key audio effects detection and auditory context inference,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 1026–1039, 2006.

[2] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, “Audio keywords generation for sports video analysis,” *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 4, no. 2, pp. 1–23, 2008.

[3] Y.T. Peng, C.Y. Lin, M.T. Sun, and K.C. Tsai, “Healthcare audio event classification using hidden markov models and hierarchical hidden markov models,” in *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, 2009, pp. 1218 –1221.

[4] A. Härmä, M. F. McKinney, and J. Skowronek, “Automatic surveillance of the acoustic activity in our living environment,” *IEEE International Conference on Multimedia and Expo*, pp. 634–637, 2005.

[5] S. Chu, S. Narayanan, and C.-C. J. Kuo, “Environmental sound recognition with time-frequency audio features,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

[6] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, “Audio context recognition using audio event histograms,” in *18th European Signal Processing Conference*, Aalborg, Denmark, 2010, pp. 1272–1276.

[7] M. Shah, B. Mears, C. Chakrabarti, and A. Spanias, “Lifelogging: Archival and retrieval of continuously recorded audio using wearable devices,” in *Emerging Signal Processing Applications (ESPA)*, 2012, pp. 99 –102.

[8] G. Wichern, Jiachen Xue, H. Thornburg, B. Mechtley, and A. Spanias, “Segmentation, indexing, and retrieval for environmental and natural sounds,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 688 –707, 2010.

[9] X. Zhou, X. Zhuang, M. Liu, H. Tang, M. Hasegawa-Johnson, and T. Huang, “HMM-based acoustic event detection with AdaBoost feature selection,” in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, Berlin, Heidelberg, 2008, pp. 345–353, Springer-Verlag.

[10] X. Zhuang, X. Zhou, M. A. Hasegawa-Johnson, and T. S. Huang, “Real-world acoustic event detection,” *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1543 – 1551, 2010, Pattern Recognition of Non-Speech Audio.

[11] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, “Acoustic event detection in real-life recordings,” in *18th European Signal Processing Conference*, Aalborg, Denmark, 2010, pp. 1267–1271.

[12] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, “Sound event detection in multisource environments using source separation,” in *Workshop on Machine Listening in Multisource Environments, CHiME2011*, Florence, Italy, 2011.

[13] T. Butko, González P., Segura C., Nadeu C., and Hernando J., “Two-source acoustic event detection and localization: Online implementation in a smart-room,” in *19th European Signal Processing Conference*, Barcelona, Spain, 2011, pp. 1317–1321.

[14] A. Temko and C. Nadeu, “Acoustic event detection in a meeting-room environment,” *Pattern Recognition Letters*, vol. 30, no. 14, pp. 1281–1288, 2009.

[15] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, “Context-dependent sound event detection,” *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2013, no. 1, 2013.

[16] Taras Butko, Cristian Canton-Ferrer, Carlos Segura, Xavier Giró, Climent Nadeu, Javier Hernando, and Josep R. Casas, “Acoustic event detection based on feature-level fusion of audio and video modalities,” *EURASIP Journal on Advanced Signal Processing*, vol. 2011, no. 1, 2011.

[17] L. Deng, A. Acero, M. Plumpe, and X. Huang, “Large-vocabulary speech recognition under adverse acoustic environments,” in *International Conference on Spoken Language Processing (ICSLP)*, 2000, pp. 806–809.

[18] T. Virtanen, “Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

PUBLICATION P5

T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, “Audio Context Recognition Using Audio Event Histograms,” in *Proceedings of 2010 European Signal Processing Conference*, (Aalborg, Denmark),pp. 1272–1276, 2010.

AUDIO CONTEXT RECOGNITION USING AUDIO EVENT HISTOGRAMS

Toni Heittola[1], *Annamaria Mesaros*[1], *Antti Eronen*[2], *Tuomas Virtanen*[1]

[1]Department of Signal Processing
Tampere University of Technology
Korkeakoulunkatu 1, 33720, Tampere, Finland
email: toni.heittola@tut.fi, annamaria.mesaros@tut.fi,
tuomas.virtanen@tut.fi

[2]Nokia Research Center
P.O.Box 100, FIN-33721, Tampere, Finland
email: antti.eronen@nokia.com

ABSTRACT

This paper presents a method for audio context recognition, meaning classification between everyday environments. The method is based on representing each audio context using a histogram of audio events which are detected using a supervised classifier. In the training stage, each context is modeled with a histogram estimated from annotated training data. In the testing stage, individual sound events are detected in the unknown recording and a histogram of the sound event occurrences is built. Context recognition is performed by computing the cosine distance between this histogram and event histograms of each context from the training database. Term frequency–inverse document frequency weighting is studied for controlling the importance of different events in the histogram distance calculation. An average classification accuracy of 89% is obtained in the recognition between ten everyday contexts. Combining the event based context recognition system with more conventional audio based recognition increases the recognition rate to 92%.

1. INTRODUCTION

Context recognition is defined as the process of automatically determining the context around a device. Information about the surroundings would enable wearable devices to provide better service to users' needs, e.g., by adjusting the mode of operation accordingly. Compared to image or video sensing, audio has certain distinctive characteristics. Audio captures information from all directions and is relatively robust to sensor position and orientation, which allows sensing without troubling the user. Audio can provide a rich set of information which can relate to location, activity, people, or what is being spoken. The acoustic ambiance and background noise characterizes a physical location, such as inside a car, restaurant, or office.

Early listening tests conducted in [1] showed that humans are able to recognize everyday auditory contexts in 70% of cases on average and confusions are mostly between contexts that have same types of prominent sound events. The study suggested that distinct sound events recognized from the auditory scene are a salient cue for human perception of audio context. However, most of the proposed context recognition systems are modeling global acoustic characteristics of the audio context rather than sound events [2, 3, 4].

In this paper, we propose a context recognition system based on detection of individual acoustic events. Our approach assumes that different contexts, such as a street or a restaurant, are characterized by the occurrence of certain sound events. Contexts are modeled with event histograms collected from annotated recordings. The proposed system is divided into two stages, sound event detection and context recognition. A sound event detection system is used to detect sound events present in the tested context and the event histogram constructed from the recognition result is matched with context models. The system is evaluated with ten contexts that may contain the same events. The overall system scheme is presented in Figure 1.

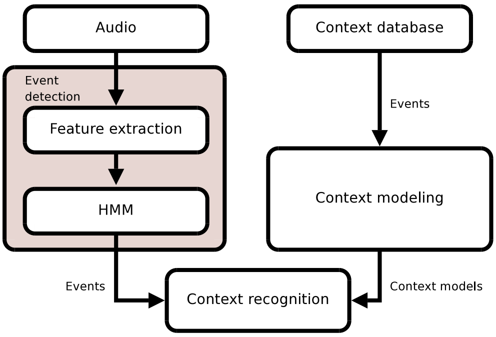


Figure 1: System overview.

The rest of this paper is organized as follows. Section 2 briefs the related work. Section 3 presents the event detection system, and Section 4 describes how detected events are used in the context recognition. Section 5 explains the context database used in the evaluation and the evaluation itself. Section 6 provides conclusions and suggestions for further study.

2. RELATED WORK

Automatic recognition of the context or environment based on audio information is known from many earlier works. However, most of the work on context recognition has been done by directly recognizing the context from the acoustic information, without explicitly detecting the individual sound events in the auditory scene. Eronen *et al.* [2] presented an approach to recognize 24 everyday context with mel-frequency cepstral coefficients (MFCC) and hidden Markov models (HMM). They reached a 58% recognition accuracy against 69% obtained in a human listening tests using the same material. The study in [3] presented an HMM-based environmental noise classification system and reported over 91% accuracy in classifying 10 contexts using three second test segments. The authors also performed a listening test on the same data. The listeners' performance for the three

[1]This work was financially supported by the Academy of Finland.

seconds segments was significantly worse than the system performance. More recently, Chu *et al.* [4] proposed an approach using matching pursuit to select a small set of time-frequency features to represent each context. They achieved a 84% performance for 14 contexts for four second segments using these features jointly with MFCC. The used contexts were chosen to be as different as possible to minimize overlapping.

One of the approaches to use sound events in the context recognition was presented in [5]. The authors propose a framework for detection of key audio effects in a continuous stream. The optimal key effect sequence is determined using Viterbi decoding, controlled by a two-loop network defining possible transitions between sound effects. They use 10 audio effects, distinct enough to be perceived, with models trained using isolated audio effects from Web. The different audio effects are modeled using HMMs with 5 to 11 states per model, trained with various features. The authors treat overlapping events by using the label of the dominant one for that region. The detected audio effects are used to recognize the scene as one of 5 possible (non-overlapping) categories - humor, pursuit, etc. More recently, the authors proposed an unsupervised co-clustering approach for the same task [6]. Authors of [7] propose an audio keywords generation system for sports videos. Low-level features are extracted from audio and after off-line feature selection hierarchical SVM is used find audio keywords. Hidden Markov models are used to detect the semantic events in sports videos. The system was tested with soccer, basketball, and tennis videos.

Sound event detection from audio signals can be performed in an unsupervised or supervised manner. In the unsupervised approach, the categories of sound events are not specified beforehand but distinct portions of the audio signal are detected as potential events, e.g. via clustering [8]. In the supervised approach, predefined sound event classes are used to segment and classify sound events. In [9], we presented a sound event detection system for the meeting room environment using MFCC based features and a HMM classifier.

3. EVENT DETECTION

The sound event detection in the proposed context recognition system is based on continuous density HMMs and the audio signal power spectrum is represented with MFCCs. These short-term features represent the coarse shape of the spectrum and provide a good discriminative performance with reasonable noise robustness. The system uses 16 MFCCs calculated from the outputs of a 40-channel filterbank. In addition to the static coefficients, their first and second order time differentials are used to describe the dynamic properties of the cepstrum. Features are extracted in 20 ms frames with a 50% frame shift.

We train 61 HMMs to represent 61 sound event categories. Three-state left-to-right HMMs are trained with the standard Baum-Welch training procedure using a training database that will be described in Section 5.1. The probability density of each state is modeled using Gaussian mixture models (GMM) having 16 components. The sound event HMMs are connected into a single HMM with equal transition probabilities between the event models.

Manually annotated recordings with overlapping events were used for training the event models. An audio segment where multiple events overlap is included in the training data of all the classes present in that segment. This means including the same observation vectors to train multiple event models. In the detection stage, features are extracted for the entire audio clip, and the event detection is organized in two ways. Event detection over the entire recording is done using the Viterbi algorithm to obtain the most likely event sequence. However, the order of the sound events will not be used in the context recognition. In addition to this, we use isolated event recognition over four second segments by finding the event HMM that has most likely produced the observation sequence of each segment. In this case, the system is used to recognize the most prominent event in each segment. A more detailed explanation of the event detection system can be found in [10].

4. CONTEXT RECOGNITION

We assume that each context is characterized by the presence of certain sound events. The event histogram for a recording is constructed by collecting all the sound events into an event occurrence histogram. In order to prevent a bias towards longer recordings, the event counts in the histogram are divided by the number of events present in the recording. The models for the contexts are constructed by summing up these event histograms. The context model histogram is normalized so that the bins sum up to one.

In the recognition stage, an event histogram is collected from the events that are detected in the tested recording. Histograms are calculated either from the output of the Viterbi segmentation or by accumulating the events recognized in the four second segments. The context recognition is based on comparing this histogram with the context histogram.

The event histograms are compared by calculating a distance between them. In the preliminary studies, we tested three distance metrics for the task: the cosine distance, the correlation distance and one based on the Kullback-Leiber divergence. Since they provided rather similar performance, in the final system we chose to use only one of them, the cosine distance. The cosine distance is defined as the cosine of the angle between an event histogram for context C and an event histogram for tested recording Q:

$$Dist_{cos}(Q,C) = \frac{\sum_{i=1}^{T} q_i c_i}{\sqrt{\sum_{i=1}^{T} q_i^2 \sum_{i=1}^{T} c_i^2}}, \qquad (1)$$

where q_i is the normalized event count of event i in the tested recording, c_i is the normalized event count of event i in the context and T is number of events in the vector. The context corresponding to the closest distance is selected as the recognition result.

In order to better model the within context variation in the distribution of events, k-nearest neighbor (k-NN) classification is also used. With k-NN, all the recordings in the training database can be used to represent the context they belong to. In this case each context is represented by several event histograms, each calculated from a single recording in the training database. Distances to each recording are calculated and the context recognition is done by majority voting among classes corresponding to the k nearest context instances.

4.1 Weighted event histograms

A weighing scheme for the events can be developed in a similar manner to the term frequency–inverse document frequency (TF-IDF) used for document indexing [11, 12]. In our case, the indexing term is the sound event and the document is a recording from a specific context or the entire context depending on the evaluation setup. The main idea of TF-IDF is that a term is an important indexing term for document d if it occurs frequently in it. This is denoted as term frequency (TF). On the other hand, terms which occur in many documents are rated less important for indexing due to their widely common nature. This is denoted as inverse document frequency (IDF) and it is defined as follows:

$$IDF(term) = log\left(\frac{|D|}{DF(term)}\right) \tag{2}$$

where $|D|$ is the total number of recordings and $DF(term)$ is the number of documents in which the term occurs at least once. The inverse document frequency of a term is low if it occurs in many documents and is highest if the term occurs in only one. The weight w_i of a term i in document d is calculated as

$$W_i = TF(term_i, d) \bullet IDF(term_i), \tag{3}$$

where $TF(term_i, d)$ is the term frequency, i.e., the number of times $term_i$ occurs in the document d.

In the training stage, IDF is collected from the training data and event histograms (TF) for contexts are weighted. In the testing stage, event histogram (TF) is collected from the test data and IDF calculated from the training data is used in the weighting of the event histogram.

5. EVALUATION

The proposed context recognition system is evaluated with an audio database collected from real-life environments. The database is used to train the event detection system and the context recognition system. Two different methods for obtaining the events are evaluated. In the first method, event recognition is done by splitting each recording into four second segments and classifying each segment as corresponding to the most likely event. The events detected in the segments within the tested recording are collected to form an event histogram. The second method uses the Viterbi algorithm to obtain the most likely event sequence for the entire recording and this sequence will be used to construct the histogram. In addition to this, two different methods for modeling each context are evaluated. The first method is to characterize each context by one histogram constructed from all the events. In the second method each recording belonging to a context is used as an example of that context and k-NN classification is used. We also study the effect of the test segment length on the recognition accuracy in detail.

5.1 Database

The material for the database was gathered by recording 10 to 30 minute long recordings in ten real-life environments or contexts. The selected audio contexts were basketball game, beach, inside a bus, inside a car, hallway, office, restaurant, grocery shop, street and stadium with track and field events. For each context, 8 to 14 recordings were made with binaural microphones placed inside the human ears. In total, 103 stereophonic recording was included in the database. In this paper, we are using monophonic versions of the recordings, i.e., two channels are averaged to one channel.

Table 1: Event statistics from the database.

Context	Number of present event classes	Total number of events	Average events per 1 min.
basketball	14	990	11.3
beach	16	738	3.7
bus	14	1729	12.0
car	12	582	5.3
hallway	9	822	7.4
office	12	1220	12.3
restaurant	13	780	7.8
shop	14	1797	20.4
street	15	827	7.6
track & field	11	793	6.9

Table 2: Context-wise average recognition performances.

	4 sec. segments	Viterbi segmentation
Cosine	88.5	84.5
TF-IDF	61.1	59.3

The recordings were manually annotated indicating the start and end times of all clearly audible sound events in the auditory scene. The repetitive sound events are usually annotated as long events, e.g. ball hitting the floor in the basketball game, while long events like conversation are annotated as multiple successive speech events if there is perceivable pause in the conversation. Annotated sound events present in the recordings were grouped into 61 event classes. The event classes include e.g. speech, laughter, applause, car door, road, dishes, door, chair, music, and footsteps. Each context contains events from 9 to 16 event classes and many event classes appear in multiple contexts. There are also event classes which are context specific. Event statistics from the recording database are presented in Table 1. Figure 2 shows the event histograms collected from the database.

The database was organized in a five-fold manner into training and testing sets, to test all the available recordings. The audio of the training set is used to train the event detection system and histograms of annotated event class occurrences are used to train the context recognition system.

5.2 Event based recognition

The results for event based context recognition are presented in Table 2. “Cosine” denotes a system were the distance between the estimated event histograms and the context histograms is calculated with the cosine distance. “TF-IDF” denotes a system were the event histograms are TF-IDF weighted before calculating the cosine distance. Two methods of collecting events are used in this evaluation. The method where event recognition is done with four second segments is denoted as “4 sec. segments” and the method using Viterbi decoding is denoted as “Viterbi segmentation” in the table.

The full confusion matrix for the system ’Cosine’ is shown in Table 3. Some of the confusions are understandable when looking at the sound events present in the contexts. For

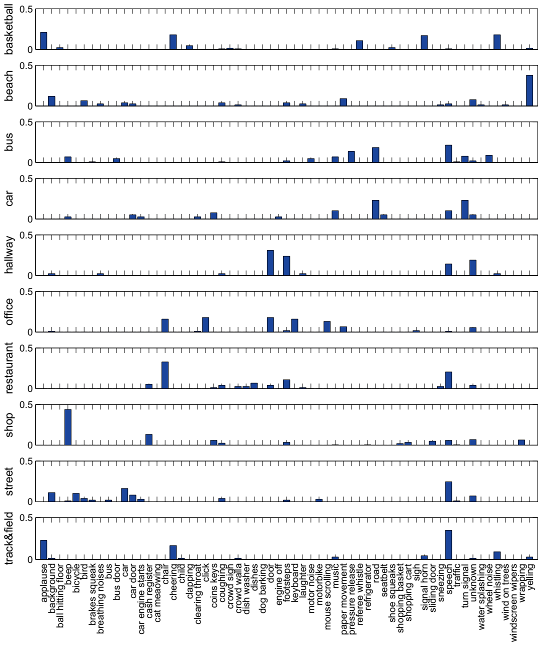


Figure 2: Normalized event histograms for contexts.

Table 3: Confusion matrix for context recognition using event histograms. Rows in the matrix correspond to presented context and columns to the recognition result.

		1	2	3	4	5	6	7	8	9	10
basketball	1	100									
beach	2		64							36	
bus	3		9	91							
car	4				100						
hallway	5					60	20	10		10	
office	6			10			90				
restaurant	7			10				90			
shop	8								100		
street	9					10				90	
track & field	10										100

example, in the hallway there are footsteps and ventilation noise present while footsteps are also present in the street context and similar ventilation noise in the office context.

Recognition results using *k*-NN approach with varying values for *k* are presented in Table 4. In this case, TD-IDF weighting helps the context recognition and provides a better performance than when using unweighted histograms. Since the idea of TF-IDF is to weigh rare events more than the common ones, collecting all the events from the database to form only one context model for each context will average out the rare events within each context and the recognition will only become more difficult.

5.3 Combining event and direct acoustic information

In addition to the event based context recognition, a system based on acoustic information of contexts was evaluated. More specifically, we constructed a baseline system where each of the ten contexts is modeled with a GMM (16 Gaussians) and using MFCCs (static, first and second order time derivatives). The test recordings for this system are cut into four second segments which are then classified individually. This system is later referred as the baseline system.

Table 4: Multiple context instances and kNN based recognition.

	$k=1$	$k=3$	$k=5$	$k=7$	$k=9$
4 second segments					
Cosine	87.3	84.6	85.8	84.8	83.8
TF-IDF	89.3	85.6	84.6	85.5	86.6
Viterbi segmentation					
Cosine	86.4	84.6	84.6	82.6	81.5
TF-IDF	89.3	87.5	87.5	89.4	89.4

Table 5: Context-wise average recognition performances.

	4 sec. segments	Viterbi segmentation
Baseline	88.5	
Cosine + Baseline	91.4	92.4
TF-IDF+ Baseline	90.5	90.4

Since the baseline system models global acoustic characteristics of the audio context instead of sound events, it may provide complementary information compared to the proposed event based system. Combining these two may thus lead to improved performance. To combine these two systems, the distance between the test event histogram and the context histograms are mapped into probabilities using an inverted sigmoid-function. The mapped probabilities are then multiplied with the context likelihood produced by the baseline system.

The evaluation results are presented in Table 5. “Baseline” denotes the system based on acoustic information of contexts and “Cosine+Baseline” denotes the system where the output of the baseline system is combined with the event based context recognition system without TF-IDF weighting. “Baseline+TF-IDF” denotes a system were the weighting of the event histograms is used. The proposed context recognition system provides comparable recognition accuracy with the baseline system (see Tables 2 and 4). The recognition accuracy is slightly improved when the proposed system is combined with the baseline system.

The full confusion matrix for the baseline system is shown in Table 6. The full confusion matrix for the system where the output of the baseline system is combined with the proposed event based system without TF-IDF weighting (see Table 3) is presented in Table 7. By comparing the confusions in Tables 6 and 7, one can see that the event based system increased the performance on the bus and hallway contexts. Confusions of the bus context are now made with the street context which is understandable since they share some sound events.

5.4 Test segment length

The effect of different test segment lengths on the recognition accuracy was evaluated. Evaluation was done by constructing the event histogram from the classification results of different number of four second segments. Using the baseline system, the likelihoods of successive four second segments are accumulated over time. The recognition results based on the test segment length are shown in Figure 3 for the baseline system and the system using *k*-NN approach.

Table 6: Confusion matrix for context recognition using the baseline system.

		1	2	3	4	5	6	7	8	9	10
basketball	1	100									
beach	2		73			9				18	
bus	3			73		9		18			
car	4				100						
hallway	5					50	30			20	
office	6					10	90				
restaurant	7							100			
shop	8								100		
street	9									100	
track & field	10										100

Table 7: Confusion matrix for context recognition using the "Cosine+Baseline" system with Viterbi segmentation.

		1	2	3	4	5	6	7	8	9	10
basketball	1	100									
beach	2		73			9				18	
bus	3			91						9	
car	4				100						
hallway	5					80	20				
office	6					10	90				
restaurant	7							100			
shop	8								100		
street	9								10	90	
track & field	10										100

5.5 Discussion

TF-IDF weighting was found to help recognition only when using multiple examples of one context, represented by the recordings in the training database. This is due to the fact that TF-IDF weights rare events more than the common ones and having only one model for the complex contexts will smooth out the rare events. Furthermore, this weighting has problems with short segments having small amount of events which are all common events, and thus will be weighted to zero.

The performance of the event based system is not superior to the baseline system. The system is more complex and requires long test segments to work properly. However, it gives complementary information (sound event labels) compared to a single context label assigned to the recording. The baseline system performs nicely with contexts which are acoustically distinguishable. Combining the event based system with the baseline system provides slightly better accuracy and robustness with acoustically similar contexts.

6. CONCLUSIONS

In this paper, event histograms were used for context recognition. Recognition was evaluated on a database consisting of 103 recordings from ten different contexts. The best recognition result, 89.4% correct, for the event based recognition was obtained using multiple context instances from the training database and a k-NN classification approach. When combining the event based context recognition with a baseline context recognition system, the performance was increased to 92.4%.

In the future, other classification methods than distance metrics and k-NN will be studied. For example, training support vector machines with the event histograms might provide better recognition results.

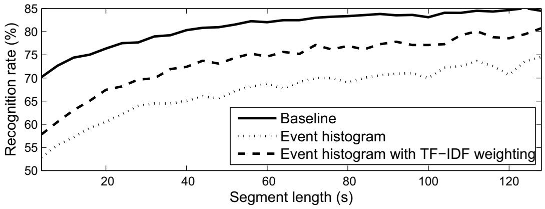


Figure 3: Context recognition accuracy as function of test segment length.

REFERENCES

[1] V. T. K. Peltonen, A. J. Eronen, M. P. Parviainen, and A. P. Klapuri, "Recognition of everyday auditory scenes: Potentials, latencies and cues," in *In Proc. 110th Audio Eng. Soc. Convention*, Hall, 2001.

[2] A. Eronen, V. Peltonen, J. Tuomi, A. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 14, pp. 321–329, Jan. 2006.

[3] L. Ma, B. Milner, and D. Smith, "Acoustic environment classification," *ACM Trans. Speech Lang. Process.*, vol. 3, no. 2, pp. 1–22, 2006.

[4] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition with time-frequency audio features," *IEEE Trans. on Audio, Speech and Language Process.*, vol. 17, no. 6, pp. 1142–1158, 2009.

[5] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L.-H. Cai, "A flexible framework for key audio effects detection and auditory context inference," *IEEE Trans. on Audio, Speech and Language Process.*, vol. 14, no. 3, pp. 1026–1039, 2006.

[6] R. Cai, L. L., and H. A., "Co-clustering for auditory scene categorization," *IEEE Trans. on Multimedia*, vol. 10, no. 4, pp. 596–606, 2008.

[7] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, "Audio keywords generation for sports video analysis," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 4, no. 2, pp. 1–23, 2008.

[8] A. Härmä, M. F. McKinney, and J. Skowronek, "Automatic surveillance of the acoustic activity in our living environment," *IEEE Int. Conf. on Multimedia and Expo*, pp. 634–637, 2005.

[9] T. Heittola and A. Klapuri, "TUT acoustic event detection system 2007," in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, pp. 364–370, Springer-Verlag, 2008.

[10] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real-life recordings," in *18th European Signal Processing Conference*, 2010.

[11] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.

[12] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for IDF," *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004.

PUBLICATION P6

A. Mesaros, T. Heittola, and T. Virtanen, “TUT Database for Acoustic Scene Classification and Sound Event Detection,” in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, pp. 1128–1132, 2016.

TUT Database for Acoustic Scene Classification and Sound Event Detection

Annamaria Mesaros, Toni Heittola, Tuomas Virtanen
Department of Signal Processing
Tampere University of Technology
Tampere, Finland
email: annamaria.mesaros@tut.fi, toni.heittola@tut.fi,tuomas.virtanen@tut.fi

Abstract—**We introduce TUT Acoustic Scenes 2016 database for environmental sound research, consisting of binaural recordings from 15 different acoustic environments. A subset of this database, called TUT Sound Events 2016, contains annotations for individual sound events, specifically created for sound event detection. TUT Sound Events 2016 consists of residential area and home environments, and is manually annotated to mark onset, offset and label of sound events. In this paper we present the recording and annotation procedure, the database content, a recommended cross-validation setup and performance of supervised acoustic scene classification system and event detection baseline system using mel frequency cepstral coefficients and Gaussian mixture models. The database is publicly released to provide support for algorithm development and common ground for comparison of different techniques.**

I. Introduction

Databases are of crucial importance in algorithm development, comparison of algorithms and reproducibility of results. Research fields that have well established benchmark databases benefit of rapid pace of development, with competition between teams on obtaining the highest performance. In this respect, detection and classification of acoustic scenes and events is picking up the pace, with special sessions organized in recent conferences and the Detection and Classification of Acoustic Scenes and Events (DCASE) 2013 challenge. This database is part of our effort to support interest in this research area and provide the research community with a starting point for data collection and common evaluation procedure.

Acoustic scene classification is defined as recognition of the audio environment, with applications in devices requiring environmental awareness [1], [2]. The environment can be defined based on physical or social context, e.g. park, office, meeting, etc. The problem is usually solved as a closed-set classification task, where identification of the current acoustic scene is required. A small number of publicly available datasets for acoustic scene classification exist. For example DCASE 2013 [3] acoustic scene development dataset contains 10 classes, 10 examples of 30 seconds length per class, with an evaluation set of the same size. Another example is the LITIS Rouen Audio scene dataset [4] containing 3026 examples for 19 classes, audio of length 30s. Additionally, a number of published studies use proprietary datasets. Results of acoustic scene classification range from 58% for 24 classes to 82% for 6 higher-level classes on the same data [2] to 93.4% for 19 classes [5]. Performance depends on the number of classes and their characteristics, with acoustic scenes that are very different from each other faring better, as expected.

Sound event detection is defined as recognition of individual sound events in audio, e.g. ”bird singing”, ”car passing by”, requiring estimation of onset and offset for distinct sound event instances and identification of the sound. Applications for sound event detection are found in surveillance, including security, healthcare and wildlife monitoring [6]–[12], audio and video content-based indexing and retrieval [13]–[15].

Sound event detection is usually approached as supervised learning, with sound event classes defined in advance and audio examples available for each class. Depending on the complexity of the required output, we differentiate between *monophonic sound event detection* in which the output is a sequence of the most prominent sound events at each time and *polyphonic sound event detection* in which detection of overlapping sounds is required [16]. Previous work on sound event detection is relatively fragmented, with studies using different, mostly proprietary datasets that are not openly available to other research groups. This hinders reproducibility and comparison of experiments. An effort in the direction of establishing a benchmark dataset was made with DCASE 2013 [3], by providing a public dataset and a challenge for different tasks in environmental sound classification. The training material contains 16 event classes, provided as isolated sound examples, 20 examples per class. The validation and evaluation data consist of synthetic mixtures containing overlapping events, 9 files for validation and 12 files for evaluation, with a length of over 1-2 minutes.

Collecting data for acoustic scene classification is a relatively quick process involving recording and annotation of audio. However, care should be taken to obtain high acoustic variability by recording in many different locations and situations for each scene class. On the other hand, annotation of audio recordings for sound event detection is a very slow process due to the presence of multiple overlapping sounds. An easier way to obtain well annotated data for sound event detection is creation of synthetic mixtures using isolated sound events - possibly allowing control of signal-to-noise ratio and amount of overlapping sounds [17]. This method has the advantage of being efficient and providing a detailed and exact ground truth. However, synthetic mixtures cannot model the variability encountered in real life, where there is no control over the number and type of sound sources and their degree of overlapping. Real-life audio data is easy to collect, but is very time consuming to annotate.

We introduce a dataset of real-life recordings that offers high quality audio for research in acoustic scene classification and polyphonic sound event detection. The audio material was carefully recorded and annotated. A cross-validation setup is provided that places audio recorded in the same location to the same side of the experiment. This avoids contamination between train and test set through use of the exact same recording conditions, which can result in over-optimistic performance through learning of acoustic conditions instead of generalization.

The paper is organized as followes: Section II introduces the data collection principles, motivating the choices made in recording, annotation and postprocessing stages. Sections III and IV present in detail TUT Acoustic Scenes 2016 - the dataset for acoustic scene classification and TUT Sound Events 2016 - the dataset for sound event detection, including statistics about their content, partitioning for system development and evaluation, and performance of a simple baseline system in a cross-validation setup on the development set. The evaluation set was later released for the DCASE 2016 challenge [18]. Finally, Section V presents conclusions and future work.

II. Data collection principles

The data collection procedure takes into account the possibility for extending this dataset by other parties, therefore it includes some rules for recording and annotation to guarantee sufficient acoustic variability and uniform labeling procedure. The sound events dataset is planned as a subset of the acoustic scene dataset, by providing specific detailed annotations of sound event instances.

A. Recording

To satisfy the requirement for high acoustic variability for all acoustic scene categories, each recording was done in a different location: different streets, different parks, different homes. High quality binaural audio was recorded, with an average duration of 3-5 minutes per recording, considering this is the most likely length that someone would record in everyday life. In general, the recording person was allowed to talk while recording, but try to minimize the amount of his own talking. Also, the recording person was required to not move much (body or head movement), to allow possible future use of spatial information present in binaural recordings. The equipment used for recording this specific dataset consists of binaural Soundman OKM II Klassik/studio A3 electret in-ear microphones and Roland Edirol R09 wave recorder using 44.1 kHz sampling rate and 24 bit resolution.

B. Annotation

Annotation of the recorded materials was done at two levels: acoustic scene annotation at recording level and detailed sound events annotation in each recording for a subset of the data. The acoustic scene categories were decided in advance.

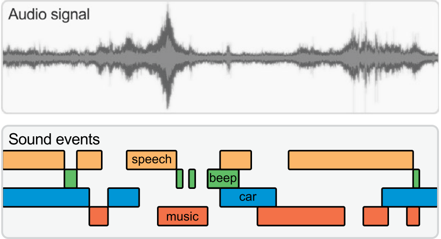


Fig. 1. Polyphonic annotation of audio.

Individual sound events in each recording were annotated using freely chosen labels for sounds. Nouns were used to characterize each sound source, and verbs to characterize the sound production mechanism, whenever this was possible. The ground truth is provided as a list of the sound events present in the recording, with annotated onset and offset for each sound instance. Sound events are overlapping, as illustrated in Fig. 1. Recording and annotation was done by two research assistants that were trained first on few example recordings. Each assistant annotated half of the data. They were instructed to annotate all audible sound events, and mark onset and offset as they consider fit. Because of the overlapping sounds, each recording had to be listened multiple times and therefore annotation was a very time consuming procedure.

C. Privacy screening and postprocessing

Postprocessing of the recorded and annotated data involves aspects related to privacy of recorded individuals, possible errors in the recording process, and analysis of annotated sound event classes. For audio material recorded in private places, written consent was obtained from all people involved. Material recorded in public places does not require such consent, but was screened for content, and privacy infringing segments were eliminated. Microphone failure and audio distortions were also annotated and this annotation is provided together with the rest of the data.

Analysis of sound event annotation reveals the diversity of the audio material. Labels for the sound classes were chosen freely, and this resulted in a large set of labels. There was no evaluation of inter-annotator agreement due to the high level of subjectivity inherent to the problem. Target sound event classes were selected based on the frequency of the obtained labels, to ensure that the selected sounds are common for an acoustic scene, and there are sufficient examples for learning acoustic models.

III. TUT Acoustic Scenes 2016

TUT Acoustic Scenes 2016 dataset consists of 15 different acoustic scenes: lakeside beach, bus, cafe/restaurant, car, city center, forest path, grocery store, home, library, metro station, office, urban park, residential area, train, and tram. All audio material was cut into segments of 30 seconds length.

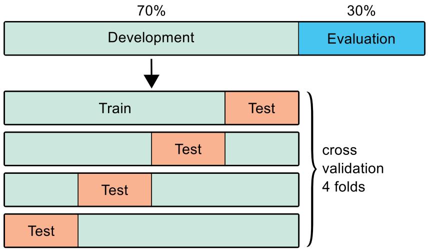


Fig. 2. Database partitioning into training and evaluation sets

A. Cross-validation setup

The dataset was split into development set and evaluation set, such that the evaluation set consists of approximately 30% of the total amount. The development set was further partitioned into four folds of training and testing sets to be used for cross-validation during system development. This process is illustrated in Fig. 2. For each acoustic scene, 78 segments were included in the development set and 26 segments were kept for evaluation.

The partitioning of the data was done based on the location of the original recordings. All segments obtained from the same original recording were included into a single subset - either development or evaluation. This is a very important detail that is sometimes neglected, and failing to recognize it results in overestimating the system performance, as the classification systems are capable of learning the location-specific acoustic conditions instead of the intended general audio scene properties,. The phenomenon is similar to the "album effect" encountered in music information retrieval, that has been noticed and is usually accounted for when setting up experiments [19]. The cross-validation setup provided with the database consists of four folds distributing the 78 segments available in the development set based on location.

B. Baseline system and evaluation

The baseline system provided with the database consists of a classical mel frequency cepstral coefficient (MFCC) and Gaussian mixture model (GMM) based classifier. MFCCs were calculated for all audio using 40 ms frames with Hamming window and 50% overlap and 40 mel bands. The first 20 coefficients were kept, including the 0th order coefficient. Delta and acceleration coefficients were also calculated using a window length of 9 frames, resulting in a frame-based feature vector of dimension 60. For each acoustic scene, a GMM class model with 32 components was trained based on the described features using expectation maximization algorithm. The testing stage uses maximum likelihood decision among all acoustic scene class models. Classification performance is measured using accuracy: the number of correctly classified segments among the total number of test segments. The classification results using the cross-validation setup for the development set is presented in Fig. 3: overall performance is 72.5%, with context-wise performance varying from 13.9% for park to 98.6% for office.

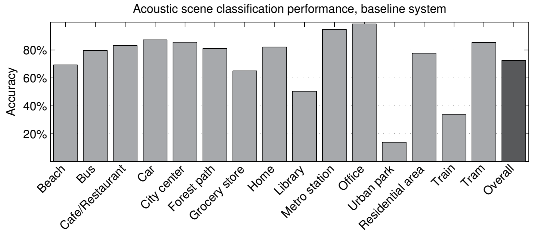


Fig. 3. TUT Acoustic Scenes 2016: Baseline system performance on development set

IV. TUT Sound Events 2016

TUT Sound Events 2016 dataset consists of two common everyday environments: one outdoor - residential area - and one indoor - home. These are environments of interest in applications for safety and surveillance (outside home) and human activity monitoring or home surveillance. The audio material consists of the original full length recordings that are also part of TUT Acoustic Scenes.

Target sound event classes were selected based on the frequency with which they appear in the raw annotations and the number of different recordings they appear in. Mapping of the raw labels was performed, merging for example "car engine running" to "engine running", and grouping various impact sounds with only verb description such as "banging", "clacking" into "object impact".

The selected event classes and their frequency are listed in Table I. It can be observed that in residential area scenes, the sound event classes are mostly related to concrete physical sound sources - bird singing, car passing by - while the home scenes are dominated by abstract object impact sounds, besides some more well defined (still impact) dishes, cutlery, etc. The provided ground truth disregards all sound events that do not belong to the target classes, despite them being present in the audio. In this respect, we provide real-life audio with annotations for selected event classes. For completeness, the detailed annotations containing all available annotated sounds are provided with the data, but the sound event detection task is planned with the event classes presented in Table I.

TABLE I
TUT Sound Events 2016: Most frequent event classes and number of instances

Residential area		**Home**	
event class	instances	event class	instances
(object) banging	23	(object) rustling	60
bird singing	271	(object) snapping	57
car passing by	108	cupboard	40
children shouting	31	cutlery	76
people speaking	52	dishes	151
people walking	44	drawer	51
wind blowing	30	glass jingling	36
		object impact	250
		people walking	54
		washing dishes	84
		water tap running	47

A. Cross-validation setup

Partitioning of data into training and evaluation subsets was done based on the amount of examples available for each event class, while also taking into account recording location. Ideally the subsets should have the same amount of data for each class, or at least the same relative amount, such as a 70-30% split. Because the event instances belonging to different classes are distributed unevenly within the recordings, we can only control to a certain extent the partitioning of individual classes. For this reason, the condition was relaxed to including 60-80% of instances of each class into the development set for residential area, and 40-80% for home. The available recordings were repeatedly randomly assigned to the sets until this condition was met for all classes.

The development set was further partitioned into four folds, such that each recording is used exactly once as test data. At this stage the only condition imposed was that the test subset does not contain classes unavailable in training. Residential area sound events data consists of five recordings in the evaluation set and four folds distributing 12 recordings into training and testing subsets. Home sound events data consists of five recordings in the evaluation set and four folds distributing 10 recordings into training and testing subsets.

B. Baseline system and evaluation

The baseline system is based on MFCCs and GMMs, with MFCCs calculated using the same parameters as in the acoustic scenes baseline system. For each event class, a binary classifier was set up. The class model was trained using the audio segments annotated as belonging to the modeled event class, and a negative model was trained using the rest of the audio. In the test stage, the decision is based on likelihood ratio between the positive and negative models for each individual class, with a sliding window of one second.

Evaluation of system performance for sound event detection uses error rate and F-score in a fixed time grid, as defined in [20]. In segments of one second length, the activities of sound event classes are compared between the ground truth and the system output. An event is considered correctly detected in a given segment if both the ground truth and system output indicate it as active in that segment. Other case are: false positive if the ground truth indicates an event as inactive and the system output indicates it as active, false negative if the ground truth indicates it as active and the system output indicates it as inactive.

Based on the total counts of true positives TP, false positives FP and false negatives FN, precision, recall, and F-score are calculated according to the formula:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F = \frac{2PR}{P + R} \tag{1}$$

Error rate measures the amount of errors in terms of *insertions* (I), *deletions* (D) and *substitutions* (S). A substitution is defined as the case when the system detects an event in a given segment, but gives it a wrong label. This is equivalent to the system output containing one false positive and one false negative in the same segment. After counting the number of substitutions per segment, the remaining false positives in the system output are counted as insertions, and the remaining false negatives as deletions. The error rate is then calculated by integrating segment-wise counts over the total number of segments K, with $N(k)$ being the number of active ground truth events in segment k [21]:

$$ER = \frac{\sum_{k=1}^{K} S(k) + \sum_{k=1}^{K} D(k) + \sum_{k=1}^{K} I(k)}{\sum_{k=1}^{K} N(k)} \tag{2}$$

Event-based metrics consider true positives, false positives and false negatives with respect to event instances. An event in the system output is considered correctly detected if it has a temporal position overlapping with the temporal position of an event with the same label in the ground truth. A collar of 200 ms was allowed for the onset, and for offset either the same 200 ms collar or a tolerance of 50% with respect to the ground truth event duration. An event in the system output that has no correspondence to an event with same label in the ground truth within the allowed tolerance is a false positive, and an event in the ground truth that has no correspondence to an event with same label in the system output within the allowed tolerance is a false negative. Event-based substitutions are defined differently than segment-based: events with correct temporal position but incorrect class label are counted as substitutions, while insertions and deletions are the events unaccounted for as correct or substituted in system output or ground truth, respectively. Precision, recall, F-score and error rate are defined the same way, with error rate being calculated with respect to the total number of events in the ground truth.

TABLE II
TUT Sound Events 2016: Baseline system performance on development set

Acoustic scene	Segment-based		Event-based	
	ER	F [%]	ER	F [%]
home	0.95	18.1	1.33	2.5
residential area	0.83	35.2	1.99	1.6
average	0.89	26.6	1.66	2.0

Performance of the baseline system on the training and development subset is presented in Table II. The results from all folds were combined to produce a single evaluation, for avoiding biases caused by data imbalance between folds [22]. While the segment-based performance is not discouraging, the performance of this baseline system evaluated using event-based metrics is extremely poor. This is easily explained by the fact that the system does not use any specific segmentation step, and it relies on the classifier to decide activity of sound classes. The binary classification scheme is not capable of detecting onsets and offsets within the evaluated tolerance. An error rate over 1.0 is also an indication of the system producing more errors than correct outputs.

A closer inspection of segment-based results reveals that there is big difference in the capability of the system to detect different classes. As can be seen in Table III, some classes are

TABLE III
TUT SOUND EVENTS 2016: SEGMENT-BASED F-SCORE CALCULATED CLASS-WISE

Residential area		**Home**	
event class	F [%]	event class	F [%]
(object) banging	0.0	(object) rustling	8.3
bird singing	33.8	(object) snapping	0.0
car passing by	59.9	cupboard	0.0
children shouting	0.0	cutlery	0.0
people speaking	30.6	dishes	4.3
people walking	2.8	drawer	8.1
wind blowing	14.2	glass jingling	0.0
		object impact	22.8
		people walking	18.3
		washing dishes	24.6
		water tap running	41.2

correctly detected in about a third of the segments, while for car passing by the detection rate is over 50%. On the other hand, the system completely fails to detect some classes. This is not surprising, considering the simplicity of the system.

V. CONCLUSIONS AND FUTURE WORK

In this paper we introduced a dataset for acoustic scene classification and sound event detection in real-world audio. The development set for both is currently available for download [23], [24], while the evaluation set will be published soon. The provided database is more complex in terms of sound event classes than previous ones, and was carefully collected to obtain a high acoustic variability of acoustic scenes. We recommend the use of the cross-validation setup for publishing future results, as this will allow exact comparison between systems. The provided cross-validation setup also ensures that all audio recorded at the same location is placed in the same subset, such that there is no data contamination between training and testing sets.

Future work will extend this data in both acoustic scenes and sound events. Other teams are invited to contribute to the dataset, by using same recording and annotation principles. The annotation procedure will be developed to improve annotation speed and and avoid ambiguity in sound event labels. Additionally, inter-annotator agreement can be used to combine the output from multiple annotators to minimize as much as possible the subjectivity of the ground truth.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Research Council under the ERC Grant Agreement 637422 EVERYSOUND.

REFERENCES

[1] D. Battaglino, L. Lepauloux, L. Pilati, and N. Evans, "Acoustic context recognition using local binary pattern codebooks," in *Worshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, October 2015.

[2] A.J. Eronen, V.T. Peltonen, J.T. Tuomi, A.P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, jan 2006.

[3] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M.D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, Oct 2015.

[4] A. Rakotomamonjy and G. Gasso, "Histogram of gradients of time-frequency representations for audio scene detection," Tech. Rep., HAL, 2014.

[5] V. Bisot, S. Essid, and G. Richard, "Hog and subband power distribution image features for acoustic scene classification," in *2015 European Signal Processing Conference (EUSIPCO)*, Nice, France, August 2015, pp. 724–728.

[6] C. Clavel, T. Ehrette, and G. Richard, "Events detection for an audio-based surveillance system," in *IEEE International Conference on Multimedia and Expo*, Los Alamitos, CA, USA, 2005, pp. 1306–1309, IEEE Computer Society.

[7] A. Härmä, M. F. McKinney, and J. Skowronek, "Automatic surveillance of the acoustic activity in our living environment," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, Los Alamitos, CA, USA, 2005, pp. 634–637, IEEE Computer Society.

[8] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Reliable detection of audio events in highly noisy environments," *Pattern Recognition Letters*, vol. 65, pp. 22 – 28, 2015.

[9] Ya-Ti Peng, Ching-Yung Lin, Ming-Ting Sun, and Kun-Cheng Tsai, "Healthcare audio event classification using hidden markov models and hierarchical hidden markov models," in *IEEE International Conference on Multimedia and Expo*, June 2009, pp. 1218–1221.

[10] S. Goetze, J. Schröder, S. Gerlach, D. Hollosi, J.E. Appell, and F. Wallhoff, "Acoustic monitoring and localization for social care," *Journal of Computing Science and Engineering*, vol. 6, no. 1, pp. 40–50, March 2012.

[11] P. Guyot, J. Pinquier, X. Valero, and F. Alias, "Two-step detection of water sound events for the diagnostic and monitoring of dementia," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, July 2013, pp. 1–6.

[12] D. Stowell and D. Clayton, "Acoustic event detection for multiple overlapping similar sources," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustic (WASPAA)*, New Paltz, NY, October 2015.

[13] R. Cai, Lie Lu, A. Hanjalic, Hong-Jiang Zhang, and Lian-Hong Cai, "A flexible framework for key audio effects detection and auditory context inference," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 3, pp. 1026–1039, May 2006.

[14] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, "Audio keywords generation for sports video analysis," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 4, no. 2, pp. 1–23, 2008.

[15] M. Bugalho, J. Portelo, I. Trancoso, T.s Pellegrini, and A. Abad, "Detecting audio events for semantic video search.," in *Interspeech*, 2009, pp. 1151–1154.

[16] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Context-dependent sound event detection," *EURASIP Journal on Audio, Speech and Music Processing*, 2013.

[17] D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M.D. Plumbley, "Detection and classification of acoustic scenes and events: An IEEE AASP challenge," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013*, Oct 2013, pp. 1–4.

[18] "Detection and Classification of Acoustic Scenes and Events 2016, IEEE AASP Challenge," http://www.cs.tut.fi/sgn/arg/dcase2016/, [Online; accessed 5-Feb-2016].

[19] Y.E. Kim, D.S. Williamson, and S. Pilli, "Towards quantifying the album-effect in artist classification," in *In Proceedings of the International Symposium on Music Information Retrieval*, 2006.

[20] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, pp. 162, 2016.

[21] G. Poliner and D. Ellis, "A discriminative model for polyphonic piano transcription," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 048317, 2007.

[22] G. Forman and M. Scholz, "Apples-to-apples in cross-validation studies: Pitfalls in classifier performance measurement," *SIGKDD Explor. Newsl.*, vol. 12, no. 1, pp. 49–57, Nov. 2010.

[23] A. Mesaros, T. Heittola, and T. Virtanen, "TUT Acoustic Scenes 2016," https://zenodo.org/record/45739, 2016, DOI: 10.5281/zenodo.45739.

[24] A. Mesaros, T. Heittola, and T. Virtanen, "TUT Sound Events 2016," https://zenodo.org/record/45759, 2016, DOI: 10.5281/zenodo.45759.

PUBLICATION P7

A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, “Detection and Classification of Acoustic Scenes and Events: Outcome of the DCASE 2016 Challenge,” in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(2):379–393, Feb 2018.

Detection and Classification of Acoustic Scenes and Events: Outcome of the DCASE 2016 Challenge

Annamaria Mesaros, Toni Heittola, Emmanouil Benetos, *Member, IEEE,* Peter Foster, *Member, IEEE,* Mathieu Lagrange, Tuomas Virtanen, *Senior Member, IEEE,* and Mark D. Plumbley, *Fellow, IEEE*

***Abstract*—Public evaluation campaigns and datasets promote active development in target research areas, allowing direct comparison of algorithms. The second edition of the challenge on Detection and Classification of Acoustic Scenes and Events (DCASE 2016) has offered such an opportunity for development of state-of-the-art methods, and succeeded in drawing together a large number of participants from academic and industrial backgrounds. In this paper, we report on the tasks and outcomes of the DCASE 2016 challenge. The challenge comprised four tasks: acoustic scene classification, sound event detection in synthetic audio, sound event detection in real-life audio, and domestic audio tagging. We present in detail each task and analyse the submitted systems in terms of design and performance. We observe the emergence of deep learning as the most popular classification method, replacing the traditional approaches based on Gaussian mixture models and support vector machines. By contrast, feature representations have not changed substantially throughout the years, as mel frequency-based representations predominate in all tasks. The datasets created for and used in DCASE 2016 are publicly available and are a valuable resource for further research.**

***Index Terms*—Acoustic scene classification, audio datasets, pattern recognition, sound event detection**

I. Introduction

Environmental sound classification and detection is a rapidly developing research area. Its growth has been stimulated by emerging public evaluation campaigns and datasets promoting active development in areas like automatic classification of acoustic scenes and automatic detection and classification of sound events. The series of challenges on Detection and Classification of Acoustic Scenes and Events (DCASE) provides a great opportunity for development and comparison of state-of-the-art methods, by offering a set of tasks with corresponding datasets, metrics and evaluation frameworks for specific topics within this research field.

Evaluation campaigns are common in many research areas and play an important role in advancing research and algorithm development. In the broad field of audio processing, automatic speech recognition evaluations have a long history [1], while the Music Information Retrieval Evaluation eXchange (MIREX) [2] has been running yearly for over a decade already. From neighboring research areas, the TRECVid Multimedia Event Detection (MED) evaluation track [3] that deals with detecting user defined events in videos, includes and encourages use of audio information for detection. Related public evaluation campaigns also include SiSEC challenge on signal separation [4] and the REVERB challenge on reverberant speech processing research [5]. Over the years, the proposed evaluation tasks in these campaigns have grown in data size, data complexity and task difficulty. In addition, evaluation campaigns that deal with more specialized topics have also appeared, for example detection of birds in audio [6].

Research in environmental sound classification and detection is part of *computational auditory scene analysis*, and is currently receiving large amounts of interest within the audio research community, manifested through special issues and sessions in related journal and conferences. The high volume of recent publications on such topics is fueled by interest in context awareness, content-based information processing of continuously growing amounts of audio material, and not least by the development of strong computational methods based on deep learning architectures. Two main research directions are evident within the computational auditory scene analysis field: acoustic scene classification as a general environment recognition problem, and sound events classification or detection as a more detailed attempt at describing the environment through the sounds encountered in it.

Acoustic scene classification is based on the premise that it is possible to provide a textual label as a general characterization of a location or situation, which is assumed to be distinguishable from others based on its general acoustic properties. The problem is typically framed as supervised classification, and often involves a relatively small number of classes. A thorough review of features and classifiers used for acoustic scene classification is presented in [7], presenting in detail the approaches submitted for DCASE 2013. Existing approaches often include use of mel-frequency cepstral coefficients and other low level spectral descriptors [8], [9] or more specialized features such as histograms of sound events [10] or histogram of gradients learned from time-frequency representations [11]. On the acoustic modeling

Manuscript received Month, Day, 2017; revised Month, Day, 2017. AM, TH and TV received funding from the European Research Council under the ERC Grant Agreement 637422 EVERYSOUND. EB is supported by a UK RAEng Research Fellowship (RF/128). MDP is partly supported by a Grant (EP/N014111/1) from the UK Engineering and Physical Sciences Research Council (EPSRC).

A. Mesaros, T. Heittola and T. Virtanen are with the Department of Signal Processing, Tampere University of Technology, 33720, Finland, e-mail: {annamaria.mesaros, toni.heittola, tuomas.virtanen}@tut.fi

E. Benetos and P. Foster are with the Centre for Digital Music, Queen Mary University of London, London E1 4NS, U.K., email: {emmanouil.benetos, p.a.foster}@qmul.ac.uk

M. Lagrange is with the ADTSI, IRCCYN, Ecole Centrale de Nantes, Nantes 44321, France.

M. D. Plumbley is with the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, U.K., email: m.plumbley@surrey.ac.uk

aspect, methods range from classical statistical models like hidden Markov models (HMMs) [8], Gaussian mixture models (GMMs) [9] or support vector machines (SVMs) [11], to more recently developed methods using deep learning that have high computational complexity in training and often have a large number of parameters [12].

Sound event detection and classification are based on the premise that sounds produced from the same source or through the same physical process can be grouped into a category, and can be distinguished from sounds originating from different sources or through different processes. In existing literature there is often not a clear distinction between detection and classification, with many early works dealing only with classification of isolated sounds. Hereafter we refer to sound event detection within an audio segment as classifying the sound into a category and locating it within the audio in terms of onset and offset relative to the entire duration. Simplified scenarios include having a single sound event per audio segment [13] or a sequence of non-overlapping sound events as the Office Live task in DCASE 2013 [6]. The most complex variant of sound event detection, referred to as *polyphonic*, involves detection of overlapping sound events. Often based on mel-scale spectral representations of the signal for features, employed methods for sound event detection include HMMs [14], NMF [15]–[17], and recently a variety of temporally constrained deep learning methods such as convolutional neural networks (CNNs) [18]–[20] and long short-term memory (LSTM) [21], [22].

As an alternative to acoustic scene classification and event detection, we may attempt to characterise an audio segment by assigning to it one or more labels, where each label indicates the presence of a particular acoustic event class in the audio segment without the need to locate the event. Thus formulating *audio tagging* as a multi-label classification task, we may consider the particular case where each training instance is an audio segment with a set of assigned labels. Since the labels provide no indication about onset and duration of acoustic events, we may consider such data weakly labeled. Whereas audio tagging has been widely applied for analyzing musical recordings [23]–[29], environmental audio tagging remains comparatively unexplored. In current studies, methods investigated include GMMs [30]–[32], SVMs combined with multiple instance learning [33], unsupervised feature learning [34], [35] and CNNs [36].

Interest for automatic environmental sound recognition has seen significant growth recently; however, in contrast to resources supporting speech or music research, databases containing environmental sounds are not easily accessible. Recently AudioSet, a large scale dataset for environmental sound research, has been made available by Google [37], containing tags for 10-second audio segments within YouTube videos; its usability in research tasks is yet to be established. Currently available literature on environmental sound recognition uses in-house datasets, making it difficult to have a fair comparison of the methods. An important step towards improving this situation was the first Detection and Classification of Acoustic Scenes and Events (DCASE) challenge organized in 2013 with purpose-built datasets. Even though the amount of data offered was rather small, the challenge introduced public evaluations of everyday sounds. DCASE 2013 was a successful first edition, covering two tasks and attracting submissions from 18 international teams, that concluded with a special session at WASPAA 2013. Thereafter, many other special sessions on environmental sound classification were organized at different conferences, marking a clear boost in research community interest in the topic.

DCASE 2016 was the second edition of the challenge, bringing the tasks closer to real life applications by using complex audio recorded in everyday life, and providing larger amounts of data for the tasks. It was organized as an IEEE Audio and Acoustic Signal Processing Technical Committee challenge, like DCASE 2013, and had a very high amount of participants overall, with four times more submissions than the first challenge. Challenge results were presented during a dedicated one day workshop. Participants came from both academia and industry, showing ongoing research and active development on both sides.

In this paper we present the tasks and outcome of the DCASE 2016 challenge, reporting advances made in the last three years. In Section II we present the DCASE 2016 Challenge organization details, timeline and tasks. We proceed with the detailed presentation of each task in Sections III–VI. For each task, we provide the definition, dataset description and experimental setup, the metrics used for evaluation of the methods, the baseline system provided to the participants as reference performance, and the analysis of submitted systems and results. Finally, Section VIII presents conclusions and provides suggestions on future work and keeping DCASE active.

II. Challenge tasks and timeline

Building on the experience from the first challenge, the tasks for DCASE 2016 were designed to improve upon those in DCASE 2013. The tasks were: Task 1 - Acoustic scene classification, Task 2 - Sound event detection in synthetic audio, Task 3 - Sound event detection in real-life audio, and Task 4 - Domestic audio tagging. Notably, Task 1 was defined the same way as in DCASE 2013, but with a new and much larger dataset, and Task 2 also considered the overlap between sound events to be detected. In addition, Task 3 was introduced to bring the challenge closer to real world applications, and Task 4 was introduced to provide a multi-label classification task.

A key difference between DCASE 2013 and 2016 is that the former asked from participating teams to submit source code for the developed systems, which was run and evaluated by the challenge organizers, thus evaluation data were not released to participants at the time. The 2016 version of the challenge instead released the evaluation data (without reference annotations) to participants, who submitted their system outputs to the challenge organizers for computation of performance metrics.

The advantage of releasing evaluation data instead of requiring source code is that it avoids potential software or output formatting incompatibilities arising from having to execute code collected from participants. For DCASE 2013,

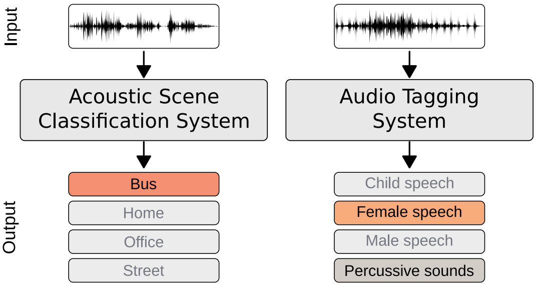


Fig. 1. Acoustic scene classification and audio tagging.

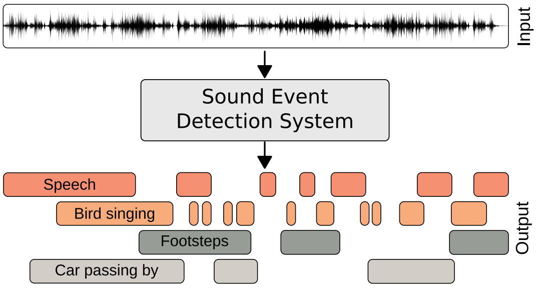


Fig. 2. Sound event detection: finding temporal positions and textual labels for sound events in an audio example.

the organizers indeed reported various software issues with libraries, Linux/Windows differences, formatting and bugs in the submitted code, which are all avoided by requiring submission of system output [38]. Given the substantially increased number of submissions for DCASE 2016, running code for all submissions would require a substantial amount of both computational and human resources. Also, requiring submission of source code may deter participants that are not comfortable or confident in their software development skills. For example the MIREX public evaluation campaign requires source code with strict rules for running it [39]; MIREX traditionally does not have many participants per task, but even so there is an imposed execution time limit, and there are cases in which the allocated execution time is exceeded. On the other hand, submission of system output to challenge organisers does not allow for a execution time analysis of submitted systems, and this practice neither actively promote good software engineering practices nor software sustainability and reproducibility. There are also potential issues with releasing datasets to participants: e.g. for MIREX several datasets are copyright-restricted (which is why they are not shared with participants), as well as on re-using datasets for several editions of a challenge.

A. *Task descriptions*

Acoustic Scene Classification is an audio classification problem that carries broad interest due to the development of context-aware devices and applications. It is a straightforward multi-class supervised classification problem in which the categories for classification are labels describing the acoustic scene. Figure 1 illustrates in the left panel the way the task is defined: for each audio example, the system must provide a single label; the system is trained using audio data labeled in the same way, with a single label per audio example.

Sound event detection is defined as the task of finding individual sound events in a test audio example by indicating onset, offset and textual labels for each sound event instance, as illustrated in Fig. 2. The sound event classes are predefined, making it a supervised learning task, with training data available for all classes. There were two sound event detection tasks in DCASE 2016, one using synthetic data generated from isolated sound event examples, for which training data were available as isolated sound examples, and the other using recordings of everyday scenes, for which training data containing overlapping sounds was provided, with manually annotated reference similar to the system output illustration. Use of synthetic data allows control over the number and relative levels of overlapping sounds, mixtures containing balanced classes and computation of performance metrics with reliable reference annotations. Real-life audio is more challenging, since real-life sound event classes are often unbalanced: some sound events may be arbitrarily rare, and manual annotations are subjective in both label and onset/offset positioning.

Audio tagging is defined as a multi-label classification problem, in which each possible label corresponds to a class of sound events which may occur in the acoustic scene, as illustrated in Fig. 1. When applied to short audio chunks, audio tagging can be viewed as a coarse-grained variant of sound event detection, where for each audio chunk the presence of a given label informs about whether events of a particular class occur in the chunk. Whereas the onset and duration prediction that we obtain in sound event detection is not requested in audio tagging, the temporal resolution imposed by the audio chunk size may nonetheless be sufficient for typical applications such as human activity monitoring, where predicting precise event boundaries is secondary to characterizing the acoustic scene. A potential practical benefit of audio tagging is the straightforward manual annotation process, which does not necessitate recording event boundaries. The task thus raises an interesting technical challenge, namely how to learn from such weakly labeled data.

As presented in detail in the following sections, each task carries its own distinct objective. Thus, the design of the datasets and the way the metrics are computed may differ, leading to the use of specific statistical significance evaluation procedures for each task.

B. *Challenge timeline and participants*

Organization of the challenge started in summer 2015 by planning the tasks, data recording and annotation process, converging to the definition of the four tasks. Once the tasks and evaluation procedure were agreed on, the challenge was announced to the community, and the organization procedure started. Table I lists the challenge timeline.

TABLE I
DCASE 2016 CHALLENGE SCHEDULE.

Phase	Time
Challenge announcement	June 2015
Data recording and annotation for tasks 1 and 3	June-Dec 2015
Definition of tasks and evaluation procedure	Sept-Nov 2015
Publication of challenge tasks	Dec 2015
Publication of development datasets and baseline systems	Jan 2016
Publication of evaluation datatsets	Apr 2016
Submission deadline	June 2016
Publication of results	Aug 2016

TABLE II
DCASE 2016 CHALLENGE SUBMISSION STATISTICS.

Task	Submissions	Teams	Authors
Acoustic Scene Classification	48	34	113
Sound Ev. Det. in Synth. Audio	10	9	37
Sound Ev. Det. in Real-Life Audio	16	12	45
Domestic Audio Tagging	8	7	23

The 2016 challenge attracted a substantially higher number of participants than the previous challenge, with a total of 82 submissions for the 4 tasks, with 48 tasks submitted for Task 1 (Acoustic Scene Classification). In comparison, DCASE 2013 comprised a total of 24 submissions from 18 teams. Table II lists statistics on the number of submissions and participants, while more detailed information for each task will be presented in the following sections.

III. ACOUSTIC SCENE CLASSIFICATION

The goal of acoustic scene classification is to classify a test recording into one of predefined classes that characterizes the environment in which it was recorded, such as ”park”, ”bus” ”home”, ”office”.

A. Dataset and experimental setup

The task used the TUT Acoustic Scenes 2016 dataset [40], consisting of recordings from 15 acoustic scenes: lakeside beach, bus, café/restaurant, car, city center, forest path, grocery store, home, library, metro station, office, urban park, residential area, train, and tramway. The acoustic scene categories were selected while planning the data recording procedure. All data were recorded in Finland. To obtain high acoustic variability for all acoustic scene categories, each recording was made in a different location: different streets, different parks, different homes. There are 15-18 locations for each acoustic scene category except office, for which there are only 13. For each recording location, a 3-5 minute long audio recording was captured. Recordings were made using a Soundman OKM II Klassik/studio A3, electret binaural microphone worn in the ears, and a Roland Edirol R-09 recorder using 44.1 kHz sampling rate and 24 bit resolution. All recorded audio material was then cut into segments of 30 seconds length.

The dataset was split into a development set and evaluation set, with the evaluation set consisting of approximately 30% of the total amount. The development set was further partitioned into four folds of training and testing sets to be used for cross-validation during system development. For each acoustic scene, 78 segments were included in the development set and 26 segments were kept for evaluation. The partitioning of the data was based on the location of the original recordings such that all segments obtained from the same original recording were included into a single subset – either development or evaluation, and within the development set into either the training or testing subset.

B. Baseline system and evaluation metric

The baseline system provided for the task [40] consists of a mel-frequency cepstral coefficient (MFCC) and Gaussian mixture model (GMM) classifier. MFCCs were calculated using 40 ms frames with a Hamming window, 50% overlap and 40 mel bands. For classification, the first 20 coefficients were kept, including the 0th order coefficient, along with delta and acceleration coefficients calculated using a window length of 9 frames. The 0th order MFCC was included in the feature vector for keeping information on the energy of the signal, which may provide discriminative information for certain scene classes. Each acoustic scene was modeled using a 16-component GMM trained using the expectation maximization algorithm. During testing, predictions were obtained using maximum likelihood classification among all available models, with likelihood accumulated over the entire test signal.

Classification performance is measured using accuracy, representing the number of correctly classified segments among the total number of test segments. The overall classification accuracy of the baseline system on the development data, obtained using the provided cross-validation setup, is 72.5%, with class-wise performance ranging from 13.9% to 98.6%. The baseline system classification accuracy on the subsequently released evaluation set is 77.2%. The baseline system is marked in the results as DCASE.

C. Challenge results

As seen in Table II, Task 1 is the most popular task of the 2016 challenge, with a total of 48 submissions from 34 different teams. Most submitted systems outperform the baseline system, which is expected, given its simplicity. Out of 48 submitted systems, 22 use deep learning (DL), and 7 teams use the binaural input or multiple combinations of the two audio channels.

Various classification approaches were used, including feed-forward neural networks, recurrent (RNN, including LSTM), convolutional (CNN), and combinations of neural networks with other techniques, specifically GMMs. SVM-based approaches account for 10 submitted systems, while ensemble classifiers are used in 10 other systems. The list of top performers is dominated by ensemble classifiers [41]–[43] and deep learning classification methods, in particular CNNs [12], [29], [44]. We also note that factor analysis methods perform well: i-vectors [41] and NMF [45] are among top performing systems, exploiting the fact that each scene is composed of multiple sources whose joint variations can be explained using latent variables. Table III summarises top-performing systems, including information on the features and

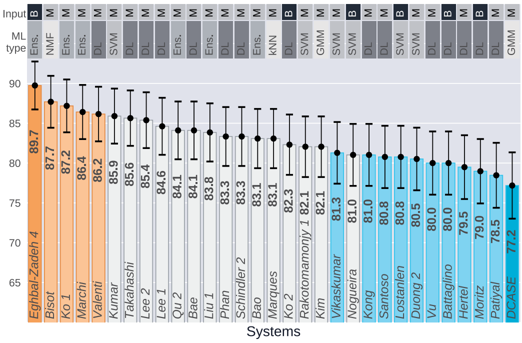


Fig. 3. Acoustic scene classification task accuracies based on the evaluation set with 95% confidence intervals, selected top system per participant. Based on McNemar's test with a significance level of 0.05, 4 runners-up systems cannot be judged to perform differently to the winner (marked in orange), and a number of systems do not perform differently to the baseline system (blue) under the same statistical test conditions.

TABLE III
SELECTED TOP RANKED SYSTEMS SUBMITTED FOR ACOUSTIC SCENE CLASSIFICATION TASK.

System	Features	Classifier	Acc
Eghbal-Zadeh 4	MFCC+spectrogram	ensemble	89.7 %
Bisot	spectrogram	NMF	87.7 %
Ko 1	various features	ensemble	87.2 %
Marchi	various features	ensemble	86.4 %
Valenti	mel energies	CNN	86.2 %
Kumar	MFCC distribution	SVM	85.9 %
Takahashi	MFCC	DNN-GMM	85.6 %
Lee 2	unsupervised features	CNN ensemble	85.4 %
Bae	spectrogram	CNN-RNN	84.6 %

classification approach employed in each system. Figure 3 lists those systems that outperform the baseline, including details on use of monophonic (M) or binaural (B) audio, and machine learning approach.

From a feature design perspective, representations using the mel-frequency scale (MFCCs and log-mel energies) were most popular among the 48 submissions. The main reason for this is that they provide a reasonably good representation of the spectral properties of the signal and provide reasonably high inter-class variability to allow class discrimination by many different machine learning approaches. Other choices included CQT-based time-frequency representations [45], combinations of various features (including mel-based) [41]–[43], and representations learned in an unsupervised way [46].

D. Discussion

Even though many of the top performing systems were based on deep learning methods, the evaluation shows that good performance can be obtained using classical methods too, such as SVM or NMF. Comparing performance between the development and evaluation datasets, most systems have similar or better performance on the evaluation set, showing that they exhibit good generalization properties.

Confidence intervals, calculated as a binomial proportion confidence interval for the classification output being correct or incorrect with respect to the ground truth, are presented in Fig. 3 for selected top systems per participant. It can be seen that the confidence intervals of systems with similar performance overlap significantly. A further analysis of the classification output using McNemar's test for comparing classifiers [47] shows that some systems cannot be considered as performing differently than the winner for a significance level of 0.05. Similarly, a number of systems cannot be differentiated from the baseline system under same statistical test conditions. Class-wise results show rather large difference in classification performance between systems and for different scene categories, with most difficult classes being library (lowest score obtained by at least one system 0%) and train (11.5%), while beach, bus, car and office had a score of at least 69% for all systems.

A listening experiment based on the evaluation dataset was set up for comparing systems' performance to human performance. Due to the size of the dataset, subsets containing 30 audio segments were presented to each test subject, with two segments for each scene class. The test segments per subject were randomly selected without replacement, resulting in the complete evaluation dataset being distributed among 13 test subjects.

A total of 87 participants provided 2 610 individual task answers. For evaluation, each audio sample is considered a separate test item and compared to the corresponding ground truth. The overall performance of the human subjects calculated over all answers was 54.4 %, while average performance across contexts for all submitted systems is 80.9% - the difference in performance is striking. Previous similar experiments resulted in human performance similar or higher than that of automatic classification methods using same data: for example in [8], human performance was 69% for 24 classes and 88% for 6 classes, just slightly higher than the automatic methods proposed in the same work; human performance for the 10 classes of DCASE 2013 data was determined to be 72% [7] and 79% [48] in two different setups, both being much better than the 55% average of the submissions. A breakdown of subjects into groups shows that the ones familiar with the Finnish soundscape had an average recognition accuracy of 60% compared to the participants from outside Finland that reached only 53%. At the same time, an expert listener who was highly familiar with the data and tested with the entire evaluation set obtained a performance of 77%.

Confusion matrices for the submitted systems and human ratings are presented in Figures 4 and 5. Some similarities can be observed, for example in the confusion of park and residential area, and train being confused with cafeteria in recordings made in the train's restaurant car. Other confusions are understandable from a human perspective, such as not distinguishing easily between forest path and park or city center and residential area streets, while another notable confusion of automatic systems is between home and library. The poor performance of humans is rather surprising, but could be explained by lack of familiarity of the subjects with the acoustic characteristics of the scene, and the small amount

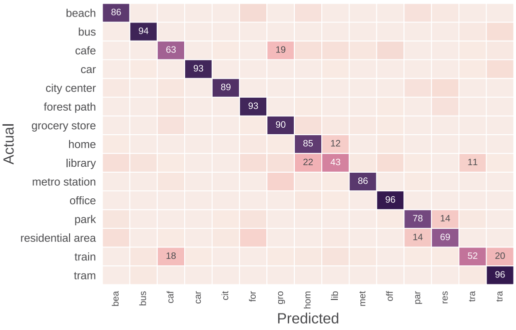


Fig. 4. Confusion matrix for all submitted systems

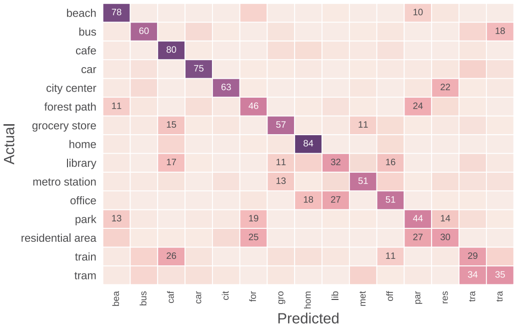


Fig. 5. Confusion matrix for human classification

of training data offered in the familiarization stage of the listening experiment. In addition, test subjects were allowed to answer whenever ready, thus reducing the amount of acoustic information used in the decision making process. A closer investigation of the listening test results is presented in [49].

IV. Sound event detection in synthetic audio

The goal of Task 2 is to detect possibly overlapping sound events, using synthetic mixtures simulating an office environment. As such, the task is directly related to the problem of *polyphonic sound event detection* and is a successor of the Event Detection - Office Synthetic task carried out at DCASE 2013 [38]. By using synthetic mixtures, Task 2 studies the behavior of tested algorithms when facing controlled levels of complexity (noise, polyphony), with the added benefit of a very accurate ground truth.

A. Dataset and experimental setup

Audio data for Task 2 contains instances of 11 sound classes related to office sounds: clearing throat, coughing, door knock, door slam, drawer, human laughter, keyboard, keys (placed on a table), page turning, phone ringing, and speech. Audio sequences for this task were created from isolated sound events using the sound scene synthesizer of [50]. Recordings of isolated sound events were made at LS2N, École Centrale de Nantes, using a shotgun microphone AT8035 connected to a ZOOM H4n recorder. Audio files are sampled at 44.1 kHz and are monophonic.

The task involves three datasets: training, development, and testing. The training set contains recordings of 20 isolated sounds per class, for the 11 classes enlisted above. The development dataset contains 18 simulated sound scenes of 2 min duration each generated using the same isolated segments found in the training dataset, plus background sounds. Finally, the test dataset contains 54 audio files of simulated sound scenes of 2 min duration each, using a pool of 440 isolated event segments not available in the training and development datasets, plus background sounds also different from the one used in the development dataset. The development and test datasets contain ground truth annotations automatically generated by the sound scene synthesizer, in the form of a sound event list identified by a start time, end time, and sound event class.

Parameters controlling the simulated material include the event-to-background ratio (EBR), the presence/absence of overlapping events (monophonic/polyphonic scene), and the number of active events per class. The EBR of an event of length N (in samples) is obtained by computing the ratio in decibel between the event E_{rms} and the background B_{rms} root mean square measures:

$$EBR = 20\log_{10}\left(\frac{E_{rms}}{B_{rms}}\right) \qquad (1)$$

where E_{rms} and B_{rms} are defined as:

$$X_{rms} = \left(\frac{1}{N}\sum_{n=1}^{N} x(n)^2\right)^{1/2} \qquad (2)$$

with $x(n)$ being replaced by either $e(n)$ or $b(n)$, the sound pressures at sample n of respectively the sound event sequence and the background noise. In the Task 2 dataset, the EBR has values of -6, 0, and 6 dB. For monophonic scenes, the number of active events per class varies from 1 to 3 and for polyphonic scenes from 3 to 5.

B. Baseline system and evaluation metrics

The baseline system developed for this task is based on supervised non-negative matrix factorization (NMF) [51] and uses a dictionary of pre-extracted spectral templates, created using the training dataset. For pre-processing, the system computes a variable-Q transform (VQT) spectrogram [52] with a 10 ms time step and a log-frequency resolution of 60 bins/octave. A simple noise removal process detects silent regions in the recording and uses them as the noise level. Supervised NMF with beta-divergence and 30 iterations is used to decompose the VQT spectrogram into a pre-extracted and fixed spectral basis matrix (estimated during training) and a sound event activation matrix. The latter matrix is subsequently thresholded and post-processed into a list of detected events per time frame.

Following a community discussion using the DCASE 2016 mailing list, a set of evaluation metrics for sound event detection was chosen. The metrics are presented in detail

TABLE IV
SUMMARY OF SYSTEMS SUBMITTED FOR THE SOUND EVENT DETECTION IN SYNTHETIC AUDIO TASK.

System	Features	Classifier	ER_{1s}	F_{1s}
Komatsu	variable Q transform	semi-supervised NMF	0.33	80.2 %
Choi	mel energy	DNN	0.36	78.7 %
Hayashi	mel filterbank	BLSTM	0.40	78.1 %
Phan	Gammatone cepstrum	Random forests	0.59	64.8 %
Giannoulis	various	CNMF	0.67	55.8 %
Pikrakis	Bark scale coefficients	Template matching	0.74	37.4 %
Vu	CQT	RNN	0.89	52.8 %
Gutierrez	MFCC	kNN	2.08	25.0 %
Kong	mel filterbank	DNN	3.54	12.6 %

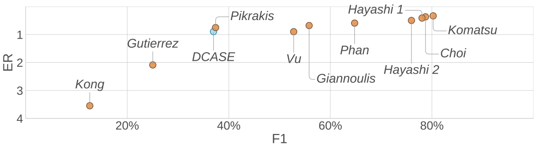


Fig. 6. Task 2 results on the evaluation set: segment-based error rate vs. F-score for all submitted systems. The baseline system is marked in blue and is ranked 8th of 11 systems.

in [53]. In Task 2, the main metric is the segment-based total error rate evaluated in one second segments over the entire test set, denoted ER_{1s}. In segment-based metrics, an event in the system output is considered correctly detected if its temporal position overlaps with the segment of an event with the same label in the ground truth. Additional metrics for Task 2 include the segment-based F-score, denoted as F_{1s}, and the onset-only event-based F-score with 200ms tolerance. Performance of the baseline system for the development dataset is $ER_{1s} = 0.78$, whereas for the test dataset $ER_{1s} = 0.89$.

C. Challenge results

Task 2 had 10 submissions from 9 teams, as can be seen in Table II. In terms of the error rate, 6 submissions outperformed the baseline system. Results in terms of the segment-based error rate and F-measure are shown in Table IV, with a graphical representation of the results shown in Fig. 6.

As can be seen from Table IV, about half of the submissions use some form of deep learning, including feedforward networks, recurrent neural networks (RNNs), bi-directional long short memory networks (BLSTMs), and convolutional neural networks (CNNs). The BLSTMs were also combined with hidden Markov models (HMMs) for modelling sound event durations. Two submissions are based on non-negative matrix factorisation (NMF), one using convolutive NMF. There were also approaches that used random forests, k-nearest neighbors and template matching. For both sets of metrics, the best performing system used NMF with a mixture of local dictionaries, combined with SVM postprocessing.

In terms of features, most approaches used time-frequency representations, including the constant-Q transform (CQT), variable-Q transform (VQT), and mel spectrograms. When compared with Task 1, the extracted features have a higher frequency resolution, in order to disambiguate multiple overlapping sound events. Other features used included MFCCs, gammatone cepstra, and Bark scale coefficients.

D. Discussion

A few submitted systems reported $ER_{1s} > 1$, which indicates that the F-measure was used as the main metric for training the submitted systems. Still, there is an almost perfect agreement with respect to rankings when comparing the error rate with the F-measure. Segment-based scores are generally higher compared to event-based scores (even considering that event-based scores only consider the sound event onset and not the offset). This drop for event-based metrics implies the lack of either temporal precision or temporal tracking in submitted systems.

With respect to the generalization capabilities of the systems, most report a significant drop in performance ($10-30\%$ in terms of absolute F-measure) when compared with development set results. This was mostly observed in conjunction with neural network-based systems, which might imply overfitting, most probably because the development set used the same samples as the training set. As expected, results depend on the sound class. For example, the first-ranked system of Komatsu et al. [54] reports an F-measure of 90.7% on door knock events, and a 37.7% on door slams. The door slam class in particular was the most challenging to detect amongst all systems, possibly due to the short duration of such events.

Due to the nature of the dataset, where groups of recordings have specific properties with respect to EBR and event density, an analysis of overall system performance for Task 2 is performed using a one-way repeated measures ANOVA. Sphericity is evaluated according to a Maulchy test [55], using a significance threshold of 0.05. This analysis, performed on the class-wise event-based F-measure, shows that out of 10 systems, 7 significantly improve upon the baseline system. This metric is chosen as it is not sensitive to the duration and density (number of events per scene) of the events.

When comparing the performance of systems to detect monophonic vs. polyphonic sequences, the ANOVA analysis does not indicate any significant difference. Results with respect to background noise show that the higher the EBR, the better is the performance of the systems (with the exception of the system of Komatsu et al [54]). Only four systems have significantly better performance than the baseline for all EBR levels. Finally, statistical significance evaluation w.r.t. the number of events does not show any influence of this parameter on the performance of the evaluated systems. A detailed statistical analysis of results for Task 2 is provided in [56].

V. Sound event detection in real-life audio

Task 3 evaluates the performance of sound event detection systems in multi-source conditions similar to everyday life, where sound sources are rarely heard in isolation. Contrary to the synthetic audio task, there is no control over the number of overlapping sound events at each time, both in the training and testing audio data.

A. Dataset and experimental setup

Task 3 uses the TUT Sound Events 2016 dataset, consisting of two common everyday environments: one outdoor (residential area) and one indoor (home). The audio material consists of the original full length recordings that are also part of TUT Acoustic Scenes with the same scene label. Target sound event classes were selected based on their frequency in the annotations. The annotations were produced by two research assistants, using nouns to characterize the sound source, and verbs to characterize the sound production mechanism, wherever this was possible. The full recording and annotation procedure is described in [40].

The event classes and the number of examples available for each class are listed in Table V. The recordings contain many other overlapping sounds, but only the listed classes are considered for the current detection task. Two sets of annotations were provided: the simplified annotations containing only the selected classes, and the full annotation containing all available annotated sounds, with the baseline system implementation based on the simplified annotation set.

The data were partitioned so that a higher proportion of instances for each class were included in the development set. This resulted in keeping 5 recordings for evaluation in each scene. The development set consists of 12 recordings for residential area having 60-80% of total available instances per class and 10 recordings for home, having 40-80% of instances. The provided cross-validation setup for the development set consists of 4 folds, in which each recording is tested exactly once.

B. Baseline system and evaluation metric

The baseline system provided for the task is based on the same method as used in Task 1. It uses MFCCs and GMMs, with MFCCs calculated using the same parameters as in the baseline system for Task 1. For each event class, a binary classifier is used, with the positive class model trained using those audio segments annotated as belonging to the modeled event class, and a negative class model trained using the remainder of the audio recording [40]. During testing, the decision for each event class is independent, based on computing the likelihood ratio between positive and negative models for the class within a one second sliding window.

Evaluation of system performance for sound event detection uses as the primary metric the segment-based error rate in one second segments, as in Task 2. Secondary metrics are the segment-based F-score and event-based error rate and F-score. The segment-based error rate of the baseline system on the development set is 0.91, while on the evaluation dataset it is 0.88. For the evaluation stage, the system was trained using the full development set, resulting in better performance due to availability of more training data.

TABLE V
TUT Sound Events 2016: Most frequent event classes with number of instances

Residential area		**Home**	
event class	instances	event class	instances
(object) banging	23	(object) rustling	60
bird singing	271	(object) snapping	57
car passing by	108	cupboard	40
children shouting	31	cutlery	76
people speaking	52	dishes	151
people walking	44	drawer	51
wind blowing	30	glass jingling	36
		object impact	250
		people walking	54
		washing dishes	84
		water tap running	47

C. Challenge results

There were 16 submissions for Task 3, originating from 12 different teams. Surprisingly, only one of the submitted systems performed better than the baseline system in terms of segment-based error rate. Systems based on deep learning accounted for most of the systems, with top 7 submissions based on DNN, RNN or fusion including deep learning architectures. Other classification approaches include random forests and one GMM-HMM solution. A system generating random events for each one second segment was also submitted, to simulate a data-driven solution tailored to the evaluation metric and using only statistics of the annotation, disregarding the audio completely. Unsurprisingly, it ranked very low.

The choice of features is dominated by mel representations: out of 16 systems, 9 use MFCCs and 4 use mel energies. The most obvious explanation for this is that MFCCs and mel energies provide a compact yet reasonably informative representation of the signal spectrum. Only one team (two submissions) exploited binaural acoustic information [57].

The segment-based performance of all submitted systems is presented in Figure 7, and top three systems according to ER are summarized in Table VI. The scatter plot in Figure 7 places the best system closest to the upper right corner. It can be noticed that 8 submissions had better F-score than the baseline system. The top system has $ER_{1s} = 0.80$, which is relatively high, considering that a zero-output system has $ER_{1s} = 1$ [53]. The F-score of the top system is however also the highest of all submissions, at 47.8 %. The runner-up in terms of ER is the baseline system, while for F-score two other submissions obtain 41.9 % and 41.1 %, respectively. Most submissions had error rates between 0.9 and 1.

D. Discussion

The trend for using deep learning is evident also for Task 3. The structure and training of neural networks allow directly

TABLE VI
SUMMARY OF SELECTED SYSTEMS SUBMITTED FOR SOUND EVENT DETECTION IN REAL LIFE AUDIO TASK.

System	Features	Classifier	ER_{1s}	F_{1s}
Adavanne 1	mel energy	RNN	0.80	47.8 %
Zoehrer	spectrogram	GRNN	0.90	39.6 %
Vu	mel energy	RNN	0.91	41.9 %
Liu	MFCC	fusion	0.92	34.5 %
Kong	MFCC	DNN	0.95	36.3 %
Pham	MFCC	DNN	0.95	11.6 %
Elizalde4	MFCC	Random Forests	0.96	33.6 %
Phan	GCC	Random Forests	0.96	23.9 %
Gorin	mel energy	CNN	0.97	41.1 %

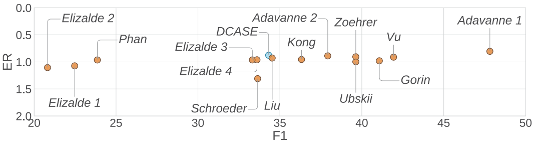


Fig. 7. Task 3 (Sound event detection in real life recordings) results using evaluation data: segment-based error rate and F-score for all submitted systems. The baseline system is marked in blue and has the second smallest error rate.

and very easily a setup for multi-label classification, which fits the task of polyphonic sound event detection. On the other hand, due to some classes having a small number of instances, most methods, and especially the deep learning methods, fail to detect them, being optimized to detect most of the events belonging to more frequent classes. A look at class-wise performance reveals that the top system detects only few classes, with F-scores 76 % for water tap and 16.5 % for washing dishes in home scenes, 62 % for bird singing, 76.7 % for car passing by, 32 % for wind blowing in residential area scenes, and all other classes 0 %. Full class-wise results are available on the challenge webpage [58].

A close look at the scene-wise performance reveals that sound events in residential area scenes were easier to detect ($ER_{1s} = 0.78$) than the ones in home scenes ($ER_{1s} = 0.91$). This is likely due to residential area classes being clearly distinct, while home classes are more similar to each other: in residential area scenes, the sound event classes are mostly related to concrete physical sound sources (bird singing, car passing by), while the home scenes are dominated by abstract object impact sounds (dishes, cutlery, etc).

Tasks 2 and 3 address the same problem and use the same metrics, but use different material (synthetic vs. real audio), resulting in a large difference in results: error rate 0.33 and F-score 80.2 % top score for synthetic data, while for real audio top scores are 0.81 and 47.8 %, respectively. This difference can be explained by the complexity of the audio: Task 2 synthetic data were generated with a controlled number of overlapping events and a quiet background, while Task 3 data have an unknown number of overlapping events, including sounds not belonging the target classes. Part of the difference in systems' performance can also stem from the manual annotation of real-world data, as manual annotations are inherently noisy and this affects both evaluation scores and training methods. Results achieved for Task 3 demonstrate the difficulty of the event detection task in a realistic setting.

VI. DOMESTIC AUDIO TAGGING

Task 4 is based on audio recordings made in a domestic environment. It involves multi-label classification of 4-second audio chunks, with the set of label classes based on prominent sound sources in the acoustic environment. For a given audio chunk, submitted systems are required to output a classification score for each of the seven label classes listed in Table VII. In the available development dataset, multi-label annotations are provided for each audio chunk.

A. Dataset and experimental setup

The audio recordings used in Task 4 originate from the Computational Hearing in Multisource Environments (CHiME) project [59], [60]. These recordings were subsequently annotated and released as CHiME-Home [32], a multi-annotation dataset aimed at audio tagging tasks.

1) Audio recordings: The CHiME-Home dataset consists of approx. 6.8 hours of stereophonic audio, obtained by positioning binaural recording equipment inside a house. The acoustic environment comprises the following sound sources: Two adults and two children, television and electronic gadgets, kitchen appliances, footsteps and knocks produced by human activity, further to sound originating from outside the house.

In Task 4, audio data are provided at sampling rates 48 kHz and 16 kHz, respectively as stereophonic and monophonic recordings. The 16 kHz recordings were obtained by downsampling the right-hand channel of the 48 kHz recordings. All audio data are available for system development, however the subsequent evaluation is performed using the monophonic audio sampled at 16 kHz. This approach aims at approximating the recording capabilities of commodity hardware.

2) Annotations: The audio was partitioned into 6 137 non-overlapping 4-second audio chunks. Subsequently, three human annotators were each asked to assign labels to each of the chunks. The set of possible label classes included those listed in Table VII, with two auxiliary labels for flagging chunks as silent or unidentifiable. To increase confidence about annotations, Task 4 is evaluated using only the chunks for which two or more annotators assigned the same label, for all considered labels. The final labels of those 2 762 chunks with 'strong agreement' between annotators are then determined by majority voting across annotators.

3) Development and evaluation data: Out of 6 137 chunks, 4 378 chunks are available for system development, with the remaining 1 759 chunks previously reserved for release after DCASE 2016, by partitioning at the level of 5-minute recording segments. The 4 378 chunks in the development dataset include 1 946 'strong agreement' chunks for training and testing. The remaining 2 432 chunks in the development dataset are available as additional training material. Out of the 1 759 chunks reserved for release after conclusion of DCASE 2016, there are 816 'strong agreement' chunks. We use these 816 chunks as evaluation data. Table VII reports label occurrences for 'strong agreement' chunks. To help quantify

TABLE VII
LABEL OCCURRENCES IN DEVELOPMENT AND EVALUATION SUBSETS OF DOMESTIC AUDIO TAGGING TASK. BASED ON AUDIO CHUNKS WITH STRONG ANNOTATOR AGREEMENT, COUNTS REPORTED IN BOLDFACE AND COUNTS WHERE ALL 3 ANNOTATORS AGREED IN ITALICS.

Label	Description	Number of audio chunks			
		Development		Evaluation	
c	Child speech	**1214**	*1143*	**328**	*301*
m	Adult male speech	**174**	*152*	**79**	*69*
f	Adult female speech	**409**	*339*	**140**	*126*
v	Video game/TV	**1181**	*1141*	**590**	*571*
p	Percussive sounds, e.g. crash, bang, knock, footsteps	**765**	*344*	**269**	*119*
b	Broadband noise, e.g. household appliances	**19**	*9*	**31**	*31*
o	Other identifiable sounds	**361**	*21*	**125**	*10*

annotator agreement, the table furthermore reports label occurrences where for a given label all 3 annotators agreed about its presence. For a discussion of annotator agreement, please refer to [32].

To aid system development, we further partition the development data at the level of 5-minute recording segments for 5-fold cross validation. In the partition, due to a low number of associated label occurrences, we omit the 5-minute recording constraint for chunks labelled ‘b’.

B. Baseline system and evaluation metric

The baseline system for Task 4 relies on MFCCs combined with GMMs. For simplicity, the system is based on the same software implementation as Task 1 and Task 3. The chosen system parameters for Task 4 closely match those previously reported for the CHiME-Home dataset [32], parameters which we observed yielded favorable results. Thus, we obtain 20 ms frames with a Hamming window and 50% overlap. Subsequently, excluding the 0th order coefficient we extract the 13 first MFCCs, based on 40 mel frequency bands. Finally, after normalizing feature vectors to zero mean and unit variance, for each of the seven considered labels we train an independent binary classifier consisting of two 8-component GMMs. Given a set of input frames, the label-wise classification score is the log-likelihood ratio of the two associated GMMs.

To quantify prediction performance with respect to a given label, we follow the convention of considering a range of possible classifier operating points. We compute the equal error rate (EER) [61], which is the fixed point of the graph of false negative rate versus false positive rate, plotted in response to the operating point. Thus, the EER approximates the classification error rate we would obtain for equal amounts of positive and negative instances, facilitating comparison of performance across label classes. Averaged across labels, the baseline system yields EERs 0.213 and 0.209, for development and evaluation datasets, respectively.

C. Challenge results

With eight submissions by seven teams, Table VIII displays obtained EERs for each of the seven individual labels, in addition to label-averaged EERs used to rank the submissions. As observed, in terms of label-averaged EERs, with the exception of two systems all submissions outperform the baseline. Obtained label-averaged EERs range from 0.166 to 0.221, with the best-performing and worst-performing submissions respectively representing a performance gain of 20.6% and a performance loss of 5.7%, relative to the baseline.

As was observed across all DCASE 2016 tasks, neural networks are a popular choice of classification technique, comprising seven out of eight submissions for Task 4. The two best-performing submissions rely on convolutional architectures. These results notwithstanding, we observe that the submission ranked third outperforms the baseline by 16.7%, while still based on GMMs. All submissions rely on widely-applied input features, with the top three submissions based on CQT features, mel spectrograms and MFCCs respectively. Across submissions, the most popular features are MFCCs.

D. Discussion

Examining label-wise EERs, averaged across submissions, we observe that the two least challenging labels are v and b, with respective mean EERs 0.061 and 0.084. Analogously averaging across submissions, the two most challenging labels are m and o, with respective mean EERs 0.267 and 0.271. The remaining labels c, p, f have the associated mean EERs 0.205, 0.218, 0.241, respectively.

As previously noted [32], a possible explanation for such variation in submission-averaged performance across labels is that perceptually salient acoustic events are relatively easy to identify: Firstly, we expect the chosen audio features to represent predominantly those events occurring in the acoustic foreground, as opposed to those events in the acoustic background. Secondly, we expect those events which occupy relatively long segments within the 4-second chunks to be readily identifiable, due to relative abundance of relevant frames for training models and building predictions. Our own informal listening suggests that sources associated with labels v and b indeed are relatively perceptually salient, frequently occupying the entire duration of audio chunks. By comparison, human utterances (labels c, m, f) are shorter in duration. Nonetheless, among human speakers, child speech appears to strongly occupy the acoustic foreground.

Table VIII indicates that the submission rankings that we obtain with respect to individual labels may deviate from the label-averaged ranking. To quantify such discrepancy between rankings, for each label we compute Spearman’s ρ between the EERs obtained for the given label, and the label-averaged EERs. Notably, we observe negative rank correlations for labels o and c, with ρ respectively -0.36 and -0.30. A possible explanation for the observed behaviour is that relevant acoustic events in chunks labelled o and c have relatively large acoustic variability are hence more prone to overfitting: For labels with large acoustic variability, we expect the relevant structure in the data to be less discernable, owing to relative data scarcity. This explanation appears consistent with the observation that the GMM-based approach submitted by Yun et al. [62] outperforms the ANN-based approaches submitted by Lidy et al. [63] and Cakir et al. [20] for label c. That the latter two submissions yield superior performance for labels m and v further suggests an advantage of ANNs combined with time-frequency input features compared to approaches based

TABLE VIII
DOMESTIC AUDIO TAGGING TASK RESULTS FOR EVALUATION DATASET, QUANTIFIED USING EQUAL ERROR RATE (EER) AND RANKED BY EER AVERAGED ACROSS LABELS.

System	Features	Classifier	Label-wise EER							Mean EER
				c	m	f	v	p	b	o
Lidy	CQT features	CNN	0.210	0.182	0.214	0.035	**0.168**	0.032	0.320	**0.166**
Cakir	Mel spectrogram	CNN	0.250	**0.159**	0.250	**0.027**	0.208	**0.022**	0.258	0.168
Yun	MFCCs	GMM	**0.177**	0.253	**0.179**	0.102	0.207	0.032	0.266	0.174
Kong	Mel spectrogram	DNN	0.195	0.280	0.229	0.090	0.221	0.039	0.272	0.189
Xu1	MFCCs	DNN	0.209	0.313	0.216	0.040	0.249	0.065	0.272	0.195
Xu2	MFCCs	DNN	0.203	0.304	0.236	0.037	0.275	0.048	0.280	0.198
DCASE	MFCCs	GMM	0.191	0.326	0.314	0.056	0.212	0.117	0.249	0.209
Vu	MFCCs	RNN	0.226	0.307	0.293	0.078	0.218	0.078	0.279	0.211
Hertel[a]	Magnitude spectrogram	CNN	0.183	0.278	0.234	0.080	0.201	0.323	**0.246**	0.221

[a] The submission by L. Hertel et al. was re-submitted after the deadline. The revised submission yielded substantially lower EERs, with the difference in performance attributed to a software bug in the original submission.

on MFCCs, for representing and identifying events occurring in the acoustic background.

To determine statistical significance of differences in performance, for each label and for each pair of submissions we apply the sign test [64] to bootstrapped paired samples of EERs. Observing that bootstrapped samples of EERs are not guaranteed to be symmetric about the median for label b, we motivate use of the sign test based on its few assumptions about underlying distributions. With the exception of the submission pair 'Vu' and 'Xu1' for label m, we observe that $p \ll 0.001$ for all combinations of submissions and labels.

VII. Discussion

At first glance, deep learning methods stand out as the most employed approach among submitted systems. The emergence of neural network based methods is also obvious in the comparison with DCASE 2013, where there were no systems involving DNNs. It is likely that besides the general popularity of deep learning as a novel technique, the amount of data available in DCASE 2016 encouraged, and to a certain extent supported their use. However, at least for the sound event detection tasks, the data size was still insufficiently large to allow robust learning. In parallel with neural networks dominating algorithm choice, it appears that data-driven approaches tend to replace manual design. The combination of these factors calls for more data, and this was seen by the participants as the main aspect that needs improvement.

The acoustic scene classification task represents the most straightforward supervised classification setup. For this reason, Task 1 attracts interest through its possible uses in applications, as well as simplicity of deploying the familiar machine learning techniques that do not require significant modifications for this task. The latter is likely the reason for which Task 1 had the highest number of participants, serving as a very good entry level task for researchers starting work in the research field. The amount of data provided for the task allowed use of deep learning algorithms involving convolutive or recurrent networks.

Sound event detection (Tasks 2 and 3) represented a more difficult setup, and this resulted into a smaller number of participants trying to tackle the problem. Participants' opinions gathered using a survey after the challenge indicate dissatisfaction with the data amount for both tasks and class balance in case of Task 3.

For Task 2 specifically, the use of simulated recordings is counterbalanced by data generated under various conditions with the benefit of a very accurate ground truth, which allowed a detailed analysis of system performance in terms of specific aspects (noise, event density, polyphony) that would hardly be possible when considering real-world data. Although we acknowledge that the sole use of simulated data cannot be considered for definitive ranking of systems, we believe that the described task design is of great interest when paired with evaluation on real world data. Despite technical improvements of the acoustic realism such as the use of reverberation filters and 3-dimensional positioning of the sources, one interesting avenue for improvements would be to design a task roughly following the evaluation procedure presented in [50]. There, systems are first evaluated against real-world data. Secondly, a synthetic dataset is designed mimicking the real-world data, ensuring that systems perform similarly. Lastly, the systems are evaluated against variants of the synthetic data. This procedure provides more grounding to the evaluation on synthetic data and can provide insights about the performance of systems for real-world data.

Naturally one should be careful in drawing conclusions obtained with simulated data only, since it is unlikely to present all the diversity present in real-world data. There are also some caveats in the use of synthetic data, that can very easily lead to erroneous conclusions. For example, one should be very careful when combining samples from multiple sample databases, since each database may have different characteristics such as audio quality, recording device, etc. In such a case, instead of recognizing target sound classes, an analysis system may learn to recognize these database-specific factors. Obviously, the same audio sample should not be present in the dataset multiple times, and definitely not in both the training and testing sets, in order to prevent overfitting. Ideally, some synthesis process could be used to produce large quantities of training material, but testing would be done with smaller amount of carefully annotated real material.

The limited amount of data available for some classes in Task 3 resulted in them not being detected at all by some systems. This indicates that the optimization process was guided by the activity of larger classes, in detriment of the classes that were insufficiently represented. However, such data are a truthful representation of real world situations, therefore in order to detect the less common sound events, systems should deal with data imbalance rather than requiring better balanced datasets.

Among the four tasks, a unique aspect of the audio tagging task is its reliance on monophonic, downsampled audio as a means of simulating the recording capabilities of commodity hardware. To better establish the effect of such audio degradation on performance, future investigations should quantify label prediction accuracy in response to a range of downsampling factors and simulated microphone characteristics.

With respect to evaluation metrics, it is worth pointing out that the current definition of segment-based metrics might not necessarily be the best way of measuring performance for sound event detection. A segment-based metric considers an event detected within a segment even if the sound is marked active for a very short duration within the segment. An event detected 10 ms early w.r.t. to reference annotation is considered correct in terms of event-based metrics, but may cause a false alarm if the 10 ms tolerance falls within the preceding segment in the segment-based metric. As a future recommendation, a rule on when an event should be considered active within a segment could be implemented, e.g. if the event is active at least 50% of the segment duration. A specific issue with the error rate is that it can result in scores surpassing 1, which can lead to interpretability issues; for this reason it is useful to also compute the F-score, to ensure that the measured output is plausible, even though it may contain many detection errors.

The area of sound scene analysis is increasingly active and there appears to be a need to maintain public evaluation campaigns such as DCASE in the foreseeable future, not only in order to evaluate the performance of state-of-the-art systems but also to serve as a focal point for the emerging research community. At the same time, issues around the long-term sustainability of the challenge need to be addressed. Based on the past challenges, a series of observations can be made:

- *Challenge organization*: Regarding central challenge organization, while DCASE 2013 and 2016 were efforts initiated by specific institutions and research groups, a direct outcome following the completion of DCASE 2016 was the establishment of a steering group comprising academics and researchers across several academic institutions and the industry, in an effort to provide high-level advice on the challenge organization. This was done in conjunction with receiving feedback from the IEEE AASP Technical Committee on Audio and Acoustic Signal Processing, as part of the committee's Challenges subgroup. A possible future direction, inspired by the MIREX challenge on music information retrieval [39] would be to de-centralize the organization of the various challenge tasks. This would enable the involvement of additional research groups and would also provide towards the long-term sustainability of the challenge by not putting too much organizational effort towards a specific research group.
- *Data collection*: Given the current setup of DCASE 2016 and the upcoming 2017 edition, where participants are given access to unlabeled test data, there is a need to produce new datasets for each new version of the challenge. Currently the reference annotations of the evaluation dataset are published after the challenge concludes. This creates sustainability issues, which can be addressed by creating artificial datasets (for example using the sound scene synthesizer of [50], as was done for DCASE 2013 and 2016), or by relaxing challenge assumptions by allowing reuse and extension of past challenge datasets.
- *Introducing new tasks*: DCASE 2013 included two tasks on sound scene classification and sound event detection, whereas the 2016 version additionally included an audio tagging task. As the field evolves, there is a need to introduce new tasks on other areas of sound scene analysis. For DCASE 2017 new tasks were introduced in the topics of rare sound event detection and weakly supervised sound event detection. As with the 'challenge organization' point above, the DCASE steering committee and its community mailing list can serve as a first point of contact towards introducing new tasks, or developing the previous tasks to make them more realistic and useful for the community.
- *Evaluation metrics*: As observed in community discussions and from results of submitted systems, there is not always an agreement on the use of evaluation metrics, which can be attributed to both disciplinary practices as well as a focus on specific applications. As part of future challenges, we note the importance of incorporating new evaluation metrics through community discussion in the DCASE mailing list, whilst however always maintaining past metrics for compatibility and completeness purposes. This was achieved as part of DCASE 2016, where the metrics toolbox for sound event detection [53] incorporated all metrics defined for DCASE 2013. In the long term, such an effort could lead to a community-led sound scene analysis evaluation toolbox similar to the 'mir_eval' toolbox for music informatics[1]. To support development and testing of new metrics, and to allow measuring performance of DCASE 2016 submissions using other metrics than the ones provided in the challenge results, the system outputs of all submitted systems were also published [2] and are available for comparison against the reference annotations.
- *Baseline systems*: A major difference between DCASE 2013 and 2016 was the introduction of a unified baseline system for Tasks 1, 3 and 4, which maintained the same back-end processing and learning methods across all tasks. Having this unified approach for all tasks can be useful to make it easier to participate in multiple tasks, and to more easily transfer findings between tasks. On the other hand, specific tasks may require substantially different techniques, which may require baselines using different learning methods. Once the research field

[1] https://github.com/craffel/mir_eval

[2] https://doi.org/10.5281/zenodo.926660

matures, we recommend that the baseline system is advanced enough, so that surpassing the baseline system performance is likely to require submitted systems to incorporate novel techniques.

VIII. Conclusions

DCASE 2016 Challenge evaluated computational methods for analysis acoustic scenes and events. Publicly available datasets, common metrics and evaluation procedures, and publicly available baseline tools allowed evaluating different algorithms independently from applications they have been developed for. The challenge was a success in terms of participation, the high number of participants showing that the topics and proposed tasks are of great importance in current audio research, and in particular on the emerging area of computational sound scene analysis. The selected tasks represent a good characterization of current interest, from the more general acoustic scene classification and audio tagging topics, to the detailed temporal detection of individual sound events.

For upcoming challenges and workshops on the topic, it is important to follow the suggestions and interest of the scientific community in the process of tasks selection, and to get involved with industrial researchers in order to have a more complete view of the research field. This will allow the community to suggest and coordinate tasks for future challenges. With the help of a steering committee comprising domain experts, the proposed tasks will be evaluated for selecting the most interesting ones and for providing feedback on their setup.

References

[1] D. S. Pallett, "A look at NIST's benchmark ASR tests: past, present, and future," in *IEEE Workshop on Automatic Speech Recognition and Understanding, 2003. ASRU'03*. IEEE, 2003, pp. 483–488.

[2] J. S. Downie, "The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247–255, 2008.

[3] G. Awad, C. G. M. Snoek, A. F. Smeaton, and G. Qunot, "TRECVid semantic indexing of video: A 6-year retrospective," *ITE Trans. on Media Technology and Applications*, vol. 4, no. 3, pp. 187–208, 2016, invited paper.

[4] N. Ono, Z. Rafii, D. Kitamura, N. Ito, and A. Liutkus, "The 2015 signal separation evaluation campaign," in *12th International Conference on Latent Variable Analysis and Signal Separation*, E. Vincent, A. Yeredor, Z. Koldovský, and P. Tichavský, Eds., 2015, pp. 387–395.

[5] K. Kinoshita, M. Delcroix, S. Gannot, E. A. P. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, A. Sehr, and T. Yoshioka, "A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, p. 7, 2016.

[6] D. Stowell, M. Wood, Y. Stylianou, and H. Glotin, "Bird detection in audio: A survey and a challenge," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept 2016, pp. 1–6.

[7] D. Barchiesi, D. Giannoulis, D. Stowell, and M. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, May 2015.

[8] A. J. Eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, Jan 2006.

[9] J.-J. Aucouturier, B. Defreville, and F. Pachet, "The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music," *The Journal of the Acoustical Society of America*, vol. 122, no. 2, pp. 881–891, 2007.

[10] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Audio context recognition using audio event histograms," in *18th European Signal Processing Conference*, Aug 2010, pp. 1272–1276.

[11] A. Rakotomamonjy and G. Gasso, "Histogram of gradients of time-frequency representations for audio scene classification," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 1, pp. 142–153, Jan. 2015.

[12] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, "DCASE 2016 acoustic scene classification using convolutional neural networks," in *Proc. of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, September 2016, pp. 95–99.

[13] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *22st ACM International Conference on Multimedia (ACM-MM'14)*, Nov. 2014.

[14] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real-life recordings," in *18th European Signal Processing Conference (EUSIPCO 2010)*, 2010, pp. 1267–1271.

[15] S. Innami and H. Kasai, "NMF-based environmental sound source separation using time-variant gain features," *Computers & Mathematics with Applications*, vol. 64, no. 5, pp. 1333 – 1342, 2012.

[16] J. Gemmeke, L. Vuegen, P. Karsmakers, B. Vanrumste, and H. Van hamme, "An exemplar-based NMF approach to audio event detection," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct 2013, pp. 1–4.

[17] A. Mesaros, O. Dikmen, T. Heittola, and T. Virtanen, "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)*, 2015, pp. 151–155.

[18] M. Espi, M. Fujimoto, K. Kinoshita, and T. Nakatani, "Exploiting spectro-temporal locality in deep learning based acoustic event detection," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 1, p. 26, 2015.

[19] H. Zhang, I. McLoughlin, and Y. Song, "Robust sound event recognition using convolutional neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[20] E. Cakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Trans. on Audio Speech and Language Processing*, 2017, arXiv preprint arXiv:1702.06286.

[21] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, and B. Schuller, "A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural network," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

[22] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 6440–6444.

[23] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green, "Automatic generation of social tags for music recommendation," in *Advances in Neural Information Processing Systems*, 2008, pp. 385–392.

[24] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 467–476, 2008.

[25] D. Tingle, Y. E. Kim, and D. Turnbull, "Exploring automatic music annotation with acoustically-objective tags," in *Proc. of the International Conference on Multimedia Information Retrieval*, 2010, pp. 55–62.

[26] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Sparse multi-label linear embedding nonnegative tensor factorization for automatic music tagging," in *18th European Signal Processing Conference*, 2010, pp. 492–496.

[27] E. Coviello, Y. Vaizman, A. B. Chan, and G. R. Lanckriet, "Multivariate autoregressive mixture models for music auto-tagging," in *International Conference on Music Information Retrieval*, 2012, pp. 547–552.

[28] K. Ellis, E. Coviello, A. B. Chan, and G. Lanckriet, "A bag of systems representation for music auto-tagging," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 21, no. 12, pp. 2554–2569, 2013.

[29] J. Lee, J. Park, K. L. Kim, and J. Nam, "Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms," *arXiv preprint arXiv:1703.01789*, 2017.

[30] B. Defréville, F. Pachet, C. Rosin, and P. Roy, "Automatic recognition of urban sound sources," in *Audio Engineering Society Convention 120*, 2006.

[31] D. Stowell and M. Plumbley, "An open dataset for research on audio field recording archives: freefield1010," in *Proc. of the AES 53rd International Conference: Semantic Audio*, 2014, pp. 80–86.

[32] P. Foster, S. Sigtia, S. Krstulovic, J. Barker, and M. D. Plumbley, "CHiME-Home: A dataset for sound source recognition in a domestic environment," in *Proc. of the 9th IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2015.

[33] A. Kumar and B. Raj, "Audio event detection using weakly labeled data," in *Proc. of the 2016 ACM on Multimedia Conference*, ser. MM '16. ACM, 2016, pp. 1038–1047.

[34] D. Stowell and M. D. Plumbley, "Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning," *PeerJ*, vol. 2, p. e488, 2014.

[35] Y. Xu, Q. Huang, W. Wang, P. Foster, S. Sigtia, P. Jackson, and M. D. Plumbley, "Unsupervised feature learning based on deep models for environmental audio tagging," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 2017.

[36] T.-W. Su, J.-Y. Liu, and Y.-H. Yang, "Weakly-supervised audio event detection using event-specific gaussian filters and fully convolutional networks."

[37] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[38] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Trans. on Multimedia*, vol. 17, no. 10, pp. 1733–1746, October 2015.

[39] "Music Information Retrieval Evaluation eXchange (MIREX)," http://music-ir.org/mirexwiki/.

[40] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, 2016.

[41] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer, "CP-JKU submissions for DCASE-2016: a hybrid approach using binaural i-vectors and deep convolutional neural networks," http://www.cs.tut.fi/sgn/arg/dcase2016/task-results-acoustic-scene-classification, DCASE2016 Challenge, Tech. Rep., September 2016, [Online; accessed 15-Mar-2017].

[42] E. Marchi, D. Tonelli, X. Xu, F. Ringeval, J. Deng, S. Squartini, and B. Schuller, "Pairwise decomposition with deep neural networks and multiscale kernel subspace learning for acoustic scene classification," in *Proc. of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, September 2016, pp. 65–69.

[43] S. Park, S. Mun, Y. Lee, and H. Ko, "Score fusion of classification systems for acoustic scene classification," http://www.cs.tut.fi/sgn/arg/dcase2016/task-results-acoustic-scene-classification, DCASE2016 Challenge, Tech. Rep., September 2016, [Online; accessed 15-Mar-2017].

[44] S. H. Bae, I. Choi, and N. S. Kim, "Acoustic scene classification using parallel combination of LSTM and CNN," in *Proc. of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, September 2016, pp. 11–15.

[45] V. Bisot, R. Serizel, S. Essid, and G. Richard, "Supervised nonnegative matrix factorization for acoustic scene classification," http://www.cs.tut.fi/sgn/arg/dcase2016/task-results-acoustic-scene-classification, DCASE2016 Challenge, Tech. Rep., September 2016, [Online; accessed 15-Mar-2017].

[46] J. Kim and K. Lee, "Empirical study on ensemble method of deep neural networks for acoustic scene classification," http://www.cs.tut.fi/sgn/arg/dcase2016/task-results-acoustic-scene-classification, DCASE2016 Challenge, Tech. Rep., September 2016, [Online; accessed 15-Mar-2017].

[47] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.

[48] J. Krijnders and G. t Holt, "Tone-fit and MFCC scene classification compared to human recognition," http://c4dm.eecs.qmul.ac.uk/sceneseventschallenge/abstracts/SC/KH.pdf, [Online; accessed 4-Apr-2017].

[49] A. Mesaros, T. Heittola, and T. Virtanen, "Assessment of human and machine performance in acoustic scene classification: DCASE 2016 case study," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, submitted.

[50] G. Lafay, M. Lagrange, M. Rossignol, E. Benetos, and A. Roebel, "A morphological model for simulating acoustic scenes and its application to sound event detection," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 24, no. 10, pp. 1854–1864, October 2016.

[51] D. D. Li and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, Oct. 1999.

[52] C. Schörkhuber, A. Klapuri, N. Holighaus, and M. Dörfler, "A Matlab toolbox for efficient perfect reconstruction time-frequency transforms with log-frequency resolution," in *AES 53rd Conference on Semantic Audio*, January 2014, p. 8 pages.

[53] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, 2016.

[54] T. Komatsu, T. Toizumi, R. Kondo, and Y. Senda, "Acoustic event detection method using semi-supervised non-negative matrix factorization with mixtures of local dictionaries," in *Proc. of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, September 2016, pp. 45–49.

[55] R. Gueorguieva and J. Krystal, "Move over ANOVA: Progress in analyzing repeated-measures data and its reflection in papers published in the archives of general psychiatry," *Archives of General Psychiatry*, vol. 61, no. 3, pp. 310–317, 2004.

[56] G. Lafay, E. Benetos, and M. Lagrange, "Sound event detection in synthetic audio: analysis of the DCASE 2016 task results," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct. 2017.

[57] S. Adavanne, G. Parascandolo, P. Pertila, T. Heittola, and T. Virtanen, "Sound event detection in multichannel audio using spatial and harmonic features," in *Proc. of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, September 2016, pp. 6–10.

[58] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. Plumbley, "Detection and classification of acoustic scenes and events DCASE2016," http://www.cs.tut.fi/sgn/arg/dcase2016/, 2016, [Online; accessed 4-Apr-2017].

[59] H. Christensen, J. Barker, N. Ma, and P. D. Green, "The CHiME corpus: a resource and a challenge for computational hearing in multisource environments," in *Proc. 11th INTERSPEECH Conf.*, 2010, pp. 1918–1921.

[60] J. Barker, E. Vincent, N. Ma, H. Christensen, and P. D. Green, "The PASCAL CHiME speech separation and recognition challenge," *Computer Speech & Language*, vol. 27, no. 3, pp. 621–633, 2013.

[61] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[62] S. Yun, S. Kim, S. Moon, J. Cho, and T. Kim, "Discriminative training of GMM parameters for audio scene classification," DCASE2016 Challenge, Tech. Rep., September 2016.

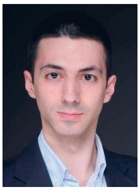
[63] T. Lidy and A. Schindler, "CQT-based convolutional neural networks for audio scene classification and domestic audio tagging," DCASE2016 Challenge, Tech. Rep., September 2016.

[64] W. J. Conover, *Practical Nonparametric Statistics*. John Wiley & Sons, 1980.

Annamaria Mesaros is a postdoctoral researcher at Laboratory of Signal Processing, Tampere University of Technology (TUT), Finland. She received the M.Sc. and Ph.D degrees in electronics and telecommunications in 2001 and 2007, respectively, from Technical University of Cluj Napoca, Romania, and Doctor of Science degree in signal processing from TUT in 2012. She has also been working as a postdoctoral researcher at Aalto University, Helsinki, Finland, within the Finnish Centre of Excellence in Computational Inference Research. Her research focuses on sound event detection in real-world multisource environments, including semantic aspects of human-generated sound annotation.

Toni Heittola received his M.Sc. degree in Information Technology from Tampere University of Technology (TUT), Finland, in 2004. He is currently pursuing the Ph.D. degree at TUT. His main research interests are sound event detection in real-life environments, sound scene classification and audio content analysis.

Emmanouil Benetos received the B.Sc. and M.Sc. degrees in informatics from the Aristotle University of Thessaloniki, Greece, in 2005 and 2007, respectively, and the Ph.D. degree in electronic engineering from Queen Mary University of London, U.K., in 2012. From 2013 to 2015, he was University Research Fellow with the Department of Computer Science, City, University of London, U.K. He is currently Lecturer and RAEng Research Fellow with the School of EECS, Queen Mary University of London, U.K. His research focuses on signal processing and machine learning for music and audio analysis, as well as applications to music information retrieval, acoustic scene analysis, and computational musicology.



Peter Foster is currently pursuing a career in industry with a focus on time series analysis. He received the Ph.D. degree from Queen Mary University of London, undertaken at the Centre for Digital Music, where he was subsequently employed as a postdoctoral research assistant. He received the M.Sc. and B.Sc. degrees in Computer Science from the University of Edinburgh and from the University of East Anglia, respectively. His interests are time series similarity and classification.

Mathieu Lagrange is a CNRS research scientist at IRCCyN, a French laboratory dedicated to cybernetics. He obtained his Ph.D. in computer science at the University of Bordeaux in 2004, and visited several institutions in Canada (University of Victoria, McGill University) and in France (Orange Labs, TELECOM ParisTech, Ircam). His research focuses on machine listening algorithms applied to the analysis of musical and environmental audio.

Tuomas Virtanen a professor at Laboratory of Signal Processing, Tampere University of Technology (TUT), Finland. He received the M.Sc. and Doctor of Science degrees in information technology from TUT in 2001 and 2006, respectively. He is known for his pioneering work on single-channel sound source separation using non-negative matrix factorization based techniques, and their application to noise-robust speech recognition, music content analysis and audio event detection. In addition to the above topics, his research interests include content analysis of audio signals in general and machine learning. He has authored about 100 scientific publications on the above topics. He is a member of the Audio and Acoustic Signal Processing Technical Committee of IEEE Signal Processing Society

Mark D. Plumbley (S'88-M'90-SM'12-F'15) received the B.A.(Hons.) degree in electrical sciences and the Ph.D. degree in neural networks from University of Cambridge, Cambridge, U.K., in 1984 and 1991, respectively. From 1991 to 2001, he was a Lecturer with Kings College London, London, U.K., before moving to Queen Mary University of London, London, in 2002, later becoming Director of the Centre for Digital Music. In 2015, he joined the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, U.K., as Professor of Signal Processing. His research interests include automatic analysis of sounds and music, including acoustic scene analysis, audio source separation, and automatic music transcription, using methods such as deep learning, matrix factorization, and sparse representations. He is a Member of the IEEE Signal Processing Society Technical Committee on Signal Processing Theory and Methods.