Temporal patterns and Ensemble Learning for Environmental Sound Recognition

by

Wenjun Yang

B.Eng, South China Agricultural University, 2015

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2017

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

Temporal patterns and Ensemble Learning for Environmental Sound Recognition

Master of Applied Science, 2017

Wenjun Yang

Electrical and Computer Engineering

Ryerson University

Abstract

This thesis explores features characterizing the temporal dynamics and the use of ensemble techniques to improve the performances of environmental sound recognition (ESR) system. Firstly, for acoustic scene classification (ASC), local binary pattern (LBP) technique is applied to extract the temporal evolution of Mel-frequency cepstral coefficients (MFCC) features, and the D3C ensemble classifier is adopted to optimize the system performance. The results show that the proposed method achieved a classification improvement of 8% compared to the baseline system.

Secondly, a new approach for sound event detection (SED) using Nonnegative Matrix Factor 2-D Deconvolution (NMF2D) and RUSBoost techniques is presented. The idea is to capture the two-dimensional joint spectral and temporal information from the time-frequency representation (TFR) while possibly separating the sound mixture into several sources. Besides, the RUSBoost ensemble technique is utilized in the event detection process to alleviate class imbalance in the training data. This method reduced the total error rate by 5% compared to the baseline method.

Acknowledgements

I would like to express my sincere appreciation to my supervisor, Dr. Sridhar Krishnan. His excellent guidance and continuous support have made my research fun and rewarding during my graduate study. He is the most erudite supervisor and gentle mentor who has encouraged me to overcome a lot of difficulties.

I am very grateful to all my fellow colleagues from the Signal Analysis Research (SAR) group for their guidance and company in and out of the lab; Alice Rueda, Jeevan Pant, Dharmendra Gurve, Sharadha Kolappan, Yashodhan Athavale and Michael Zara.

And last but not least I would like to thank my family and friends for their love and support that encouraged me from the very beginning to do my best to steadily step forward to obtain my goal.

Contents

List of Tables

List of Figures

List of Appendices

Chapter 1

Introduction

1.1 Motivation

How to make machines have the abilities to make sense of the environment remains an important research topic in several engineering and computer science disciplines. If machines are made to be aware of their surroundings as human beings are, they would be able to produce an accurate response and facilitate our lives. While our understanding of vision problems is well developed, the rest of the senses have not been investigated as much as vision has. Actually, sound also carries useful information that can be used in robot navigation, surveillance system and so on. Apart from that, it can serve as the complement to modalities such as video. Although computers can sort of recognize speech, apparently that is not all hearing is good for and we do more with our ears than just hear other people talk. Therefore, for computers to sense their environment in a human-like way, recognizing general sounds in daily environment is an important task. However, the technical challenges in this problem are plenty, since the acoustic surroundings can be quite complex. In realistic environments, the acoustic signal reaching our ears may be a complex mixture that consists of sound waves from multiple sources and in reverberant conditions.

This challenging task is defined as machine hearing by Lyon [1]. An ideal machine-hearing system would carry out analogous task to human auditory system, which means it will face a large variety of hearable sounds, and should handle all of them successfully. Because of the diversity of sound sources and application areas, typically the machine hearing task can be divided into some specific problems

based on the nature of the acoustic signal. In this way, the system becomes easier to be developed and optimized. By focusing on speech signals that can be characterized by their spectral distribution and unique phonetic structure, plenty of works in the literature deal with tasks such as speech recognition [2] and speaker identification [3]. Besides speech, music is another type of structured acoustic signal that has a set of distinctive traits. Systems designed for specific tasks such as music genre classification [4], instrument recognition [5], and music annotation [6] have been developed by researchers.

As we can see, machine hearing research has primarily been focused on speech and music signals that have some unique characteristics, and there are many theories about how these signals are processed in unique ways by our brain [7]. In contrast, other kind of sounds, referred as environmental sounds (e.g., traffic noise, door knocks, crowds.), do not exhibit such uniqueness. General environmental sounds, such as that of a thunder or a storm, have neither apparent sub-structures such as phonemes, nor meaningful stationary patterns such as melody and rhythm. Nevertheless, environmental sounds indeed contain many contextual cues that help us to identify important aspects of our surroundings, so these natural and in-the-wild sounds should also be detected and recognized by the machine hearing systems. Compared to other field such as image processing and speech processing, there is relatively little research activity for the environmental sound analysis. This type of sound can be easily collected with many devices such as mobile phone, and MP3 player without much impact to the individuals. Therefore it may have many potential applications.

1.2 Overview

1.2.1 Sound Categories

A sound taxonomy that divides sounds into several categories is given in Figure 1.1. We can see from this figure that hearable sounds are separated into five subsets, and examples describing each class are also given. In this taxonomy, only the music and speech class are well structured, other categories such as artificial and natural sounds have not been clearly defined.

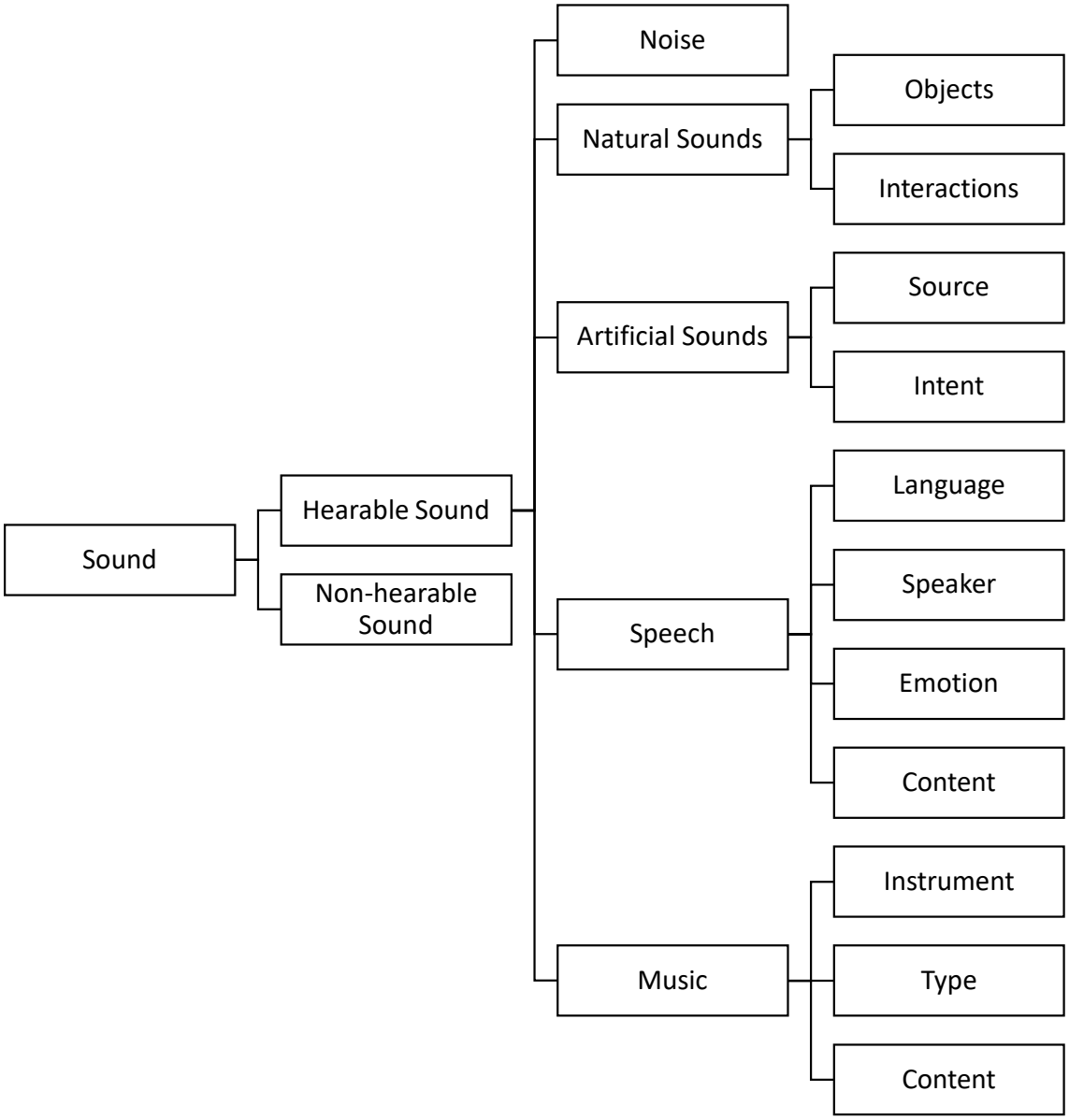


Figure 1.1: A taxonomy for sound, adapted from [8].

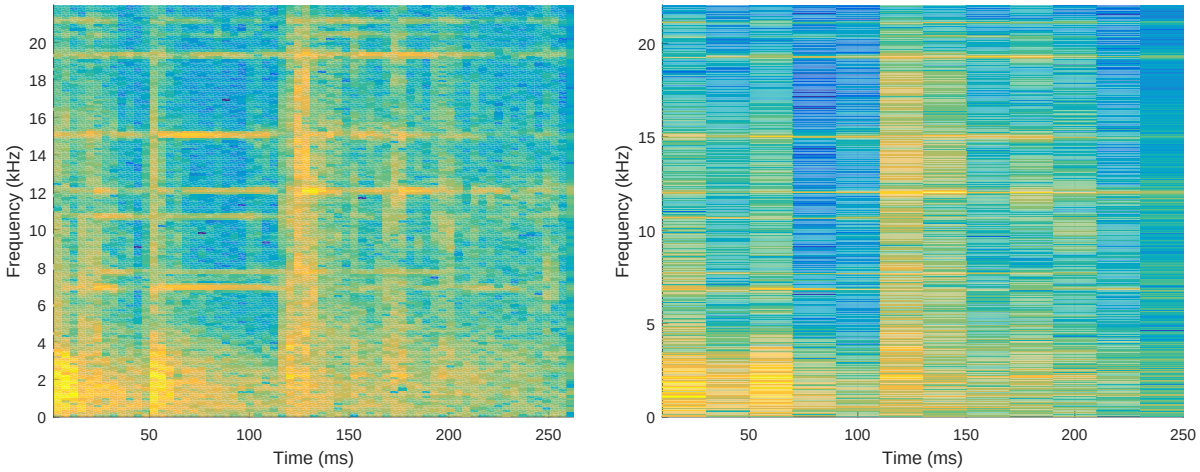
Characteristics	Speech	Music	Environmental sounds
Possible Sources	Finite	Finite	Infinite
Basic Units	Phonemes	Notes	Undefined
Stationarity	Stationary	Mostly stationary	Stationary Non-stationary
Harmonic Structure	Clear	Clear	Unclear
Length of Analysis Window	Short	Long	Undefined
Length of Shift	Short	Long	Undefined
Bandwidth	Narrow	Relatively narrow	Broad

Figure 1.2: Characteristics of speech, music and environmental sounds, adapted from [9].

1.2.2 What are Environmental Sounds?

Speech, music and environmental sounds are not necessarily mutually exclusive, since those two can occur in the environment. In this thesis, environmental sounds are used to describe naturally occurring sounds that one encounters in daily life. Sound sources are considered to be the objects to generate the sounds, and sound events refer to things or sources that may yield acoustic waveform. Therefore, events may have closely associated sounds, and that's why a listener can recognize an event or a source.

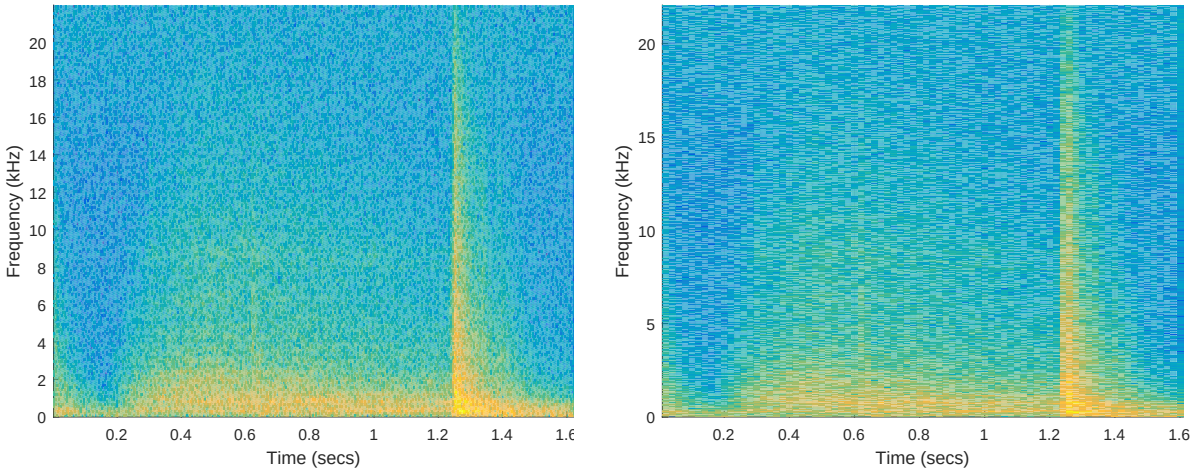
Compared to speech and music, the characteristics of environmental sounds are fairly varied and hard to generalize. It results from the wide range of sources that environmental sounds may contain. An example that compares the characteristics of speech, music and environmental sounds are listed in Figure 1.2 [9]. It can be seen that the characteristics of environmental sounds do not have clear definitions.



(a) Using short window (8msec)

(b) Using long window (40msec)

Figure 1.3: Spectrograms of key drop



(a) Using short window (8msec)

(b) Using long window (40msec)

Figure 1.4: Spectrograms of drawer

However, for speech, at least we know that speech tend to have a narrow bandwidth because most of the energy are contained in the lower frequency bands. For music, the energy may spread over a wider frequency range, leading to a broad bandwidth. Since most of the basic information about environmental sounds is unknown to us, it is more difficult to find features that characterize them well. That's why the ESR system may be task-specific, focusing on certain types of sounds.

In Figures 1.3 and 1.4, the spectrograms of the sound event key drop and drawer are shown respectively. They were analyzed using short time Fourier transform (STFT) with different window length. As we can see, for the key drop sound, the signal spectra has higher variation when using shorter windows. While, the temporal variation in spectrograms of drawer sound remains almost unchanged when using different window lengths. This may explain why classic features for speech and music analysis are not always enough for analyzing environmental sounds [10].

Another factor that makes the ESR problem challenging is the sounds reaching our ears may be a mixture of multiple overlapping sound events, which may produce features that are difficult to be identified. Besides, the noise present in the environmental sounds can also affect the system performance, as the definition of noise is relatively subjective. Even in the same acoustic scene, the noise may change with the location, weather, object, etc. The collecting device can also influence the quality of environmental sounds. Compared to speech which is recorded close to the microphone, environmental sounds can be recorded either close or far from the microphone, which makes the quality of the audio vary. Due to the application requirement, the recording of environmental sounds may be conducted on a portable and embedded device, which makes it hard to control the quality of the audio.

1.2.3 Research Tasks

There are two independent but relatively general types of task that a machine hearing system would carry out. One is the recognition of the general environment type, which is referred as ASC, and the other is the detection and classification of events occurring within a scene, which is referred as SED.

Acoustic Scene Classification

ASC refers to the task of assigning a semantic label to an audio stream that identifies the environment in which it has been produced [11]. An overview of the SED system is shown in Figure 1.5. An overview of the SED system is shown in Figure 1.5. It can be seen that ASC system performs a multi-class

classification task, in which a set of pre-defined classes are given and the system must select one as the classifcation output. Similar to many classification problem, the performance of such a system would be depended on the extracted features as well as the classification technique.

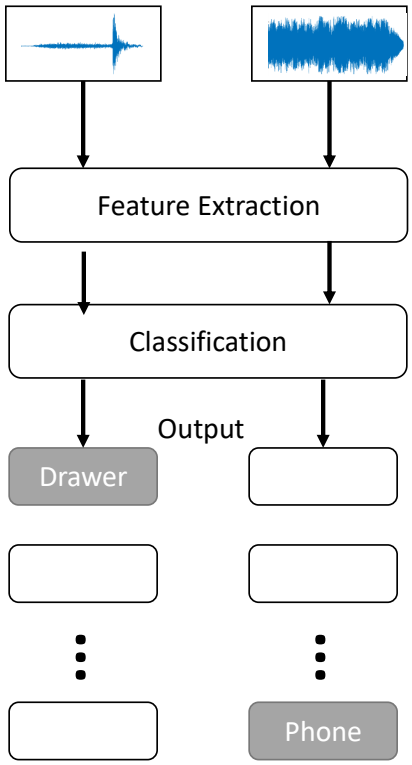


Figure 1.5: Overview of ASC system

Sound Event Detection

Different from ASC, SED refers to the task of labeling temporal regions of the active event within the audio, by predicting the start and end point of each event. Obviously, this task is more complicated than the classification one, since the detection task needs to discriminate the event categories of interest from the rich background sounds. An overview of the SED system is shown in Figure 1.6.

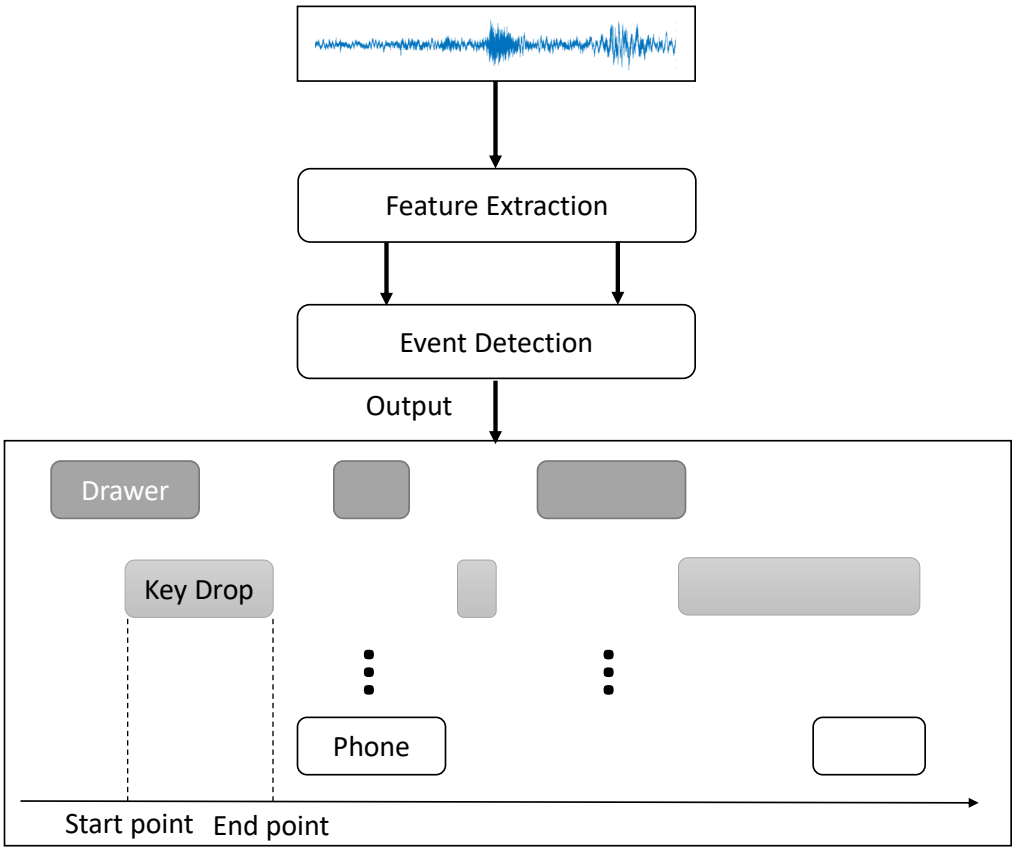


Figure 1.6: Overview of SED system

1.3 Applications

There are practical applications that specifically benefit from environmental sounds research. Examples of possible future technologies that are related to ESR include smart devices that continuously sense the environments, switching among different modes; assistive technologies such as hearing aids that adjust their setting based on the sense of the surroundings; or sound archives that automatically assign metadata to audio files. Moreover, classification of acoustic scene could be considered as a preprocessing step to adapt algorithms developed for other applications.

1.3.1 Context-aware Services

Comparing with traditional computers that only follow the users' orders, we may prefer a more smart system that can sense some context information to adapt its operation accordingly and automatically. This may be referred as Context-aware services [12], which is a computing technology that incorporates information about the current location of a user to provide a series related services to the user. Apparently, context-awareness is quite crucial to the development of such systems. Context can refer to real-world characteristics, which may include weather, time, location, recent event. If the devices are able to identify the scene and update this information by analyzing the environmental sounds, it can predict what the user may need and adjust its mode of operation for the user.

1.3.2 Intelligent Wearable Devices

Facilitated by the fact that many portable devices have built-in microphones, the device can make recordings and analyze the environmental sounds, therefore make the device intelligent. Hearing aid systems can also benefit from ESR [13]. A hearing aid is a device designed to improve hearing. As the environment information can be used to tune the parameters and settings of the hearing aid devices, the digital hearing aids can help the users hear clearly in almost any environment. The hearing aids can adjust the volumes automatically after they learn the user's preferences in different environments, and switch programs for different situations based on the analysis of environmental sounds. Therefore, the users can gain better listening experience.

1.3.3 Audio Archive Management

Archives of audio can also utilize the ESR technology [14]. The acoustic information obtained from environmental sound analysis can provide great convenience for material searching in audio libraries, especially for those big libraries where access is really time and labor consuming. Although the use of keywords can help with retrieving, this method still is subjective to each individual. If the type of acoustic scene of an audio can be recognized by the system, then it can be classified into the corresponding archives. Further, the detected sound event types can be used as keywords and saved in the metadata of the audio file. With the help of ESR, it is more efficient to browse an audio database.

1.3.4 Robotics Navigation Systems

Robotics navigation system that works in the acoustic environment is designed for the situation where the target is not inside the scope of the visual sensor [15]. In this case, only the auditory sensor can be used to detect the target. For such a system, the input audio obtained from the auditory sensor is firstly used to localize the target. When the target is detected, the decision making unit will utilize the environment information to plan a safe path to the target.

1.4 Original Contribution

In this thesis, two novel ESR methods are proposed which are expanded upon on chapters three and four. The inspiration for this comes from the idea that temporal characteristics of environmental sounds can be used to identify different sound events. Besides, the application of ensemble classifier is to produce a more robust model by combining the outputs of some base learners. With these two perspectives, the following new methods are developed to address two general tasks of ESR:

The first contribution is that we develop a new method for ASC. This novel approach utilizes temporal information of environmental sounds and the D3C ensemble classifier to improve the classification performance. We apply an image processing technique called LBP to extract the temporal dynamics of the signaland combine these features with the commonly used MFCC features. The experiments on the Detection and Classification of Acoustic Scenes and Events (DCASE) database show that this technique can outperform the baseline system.

The second contribution is that we present a new solution for SED, which captures the joint spectral and temporal information of the signal and adopts the RUSBoost ensemble technique to reduce the detection error. This work utilizes a matrix factorization method called non-negative matrix factor 2-D deconvolution (NMF2D) to generate the time-frequency templates and their corresponding activation. Three known TFR are compared and discussed to select a more effective representation. The experiments on the DCASE database show that this technique can reduce the detection error and improve the F-score when compared with the baseline system.

1.5 Thesis Organization

The organization of the thesis is shown in Figure 1.7. The remainder of this thesis is organized as follows:

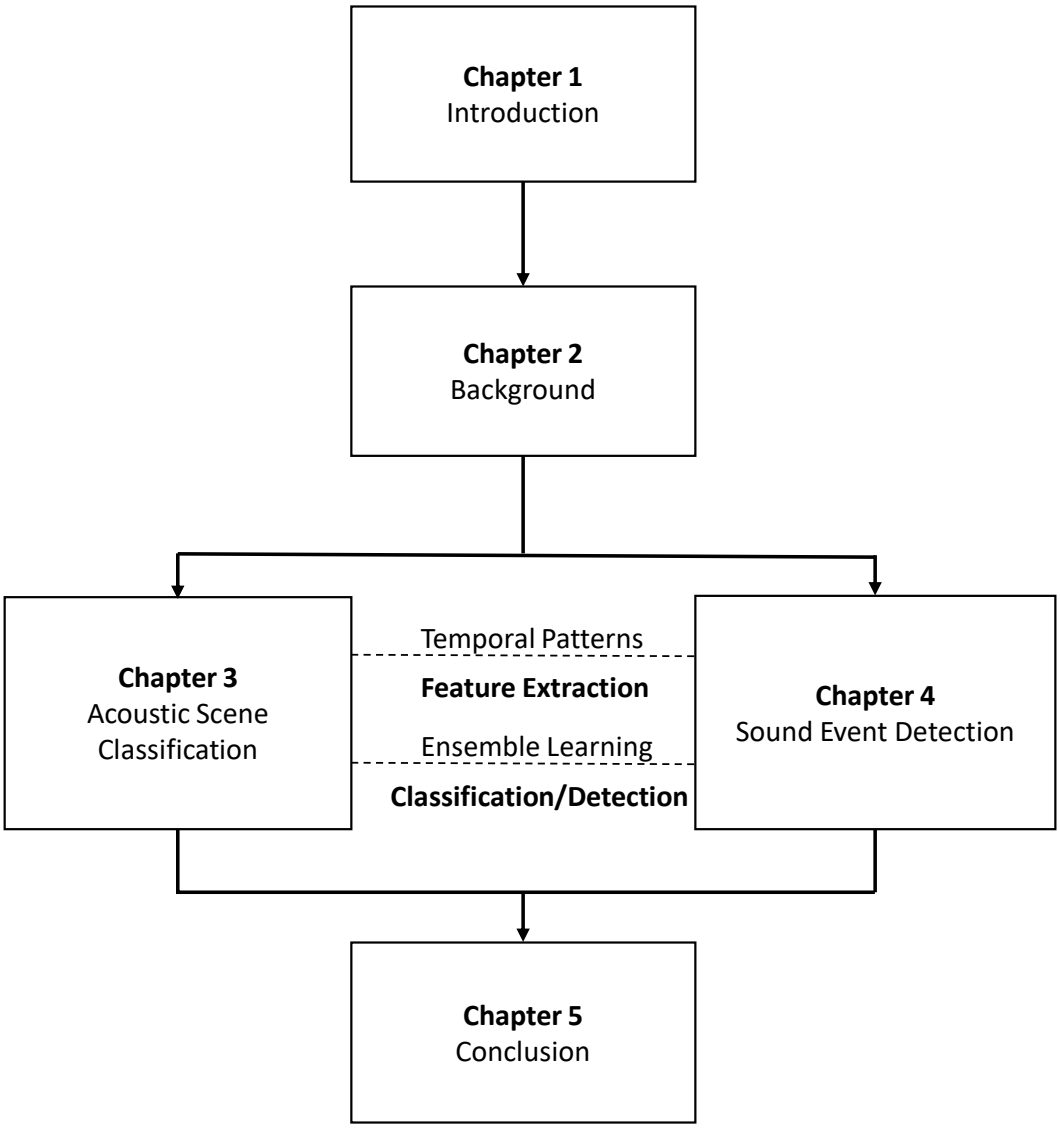


Figure 1.7: Organization of the thesis.

Chapter 2, Background, is a literature review of techniques for ASC and SED. It summarizes the commonly used acoustic features for describing environmental sounds, and the classifiers and detection schemes for the classification/detection task.

Chapter 3, Acoustic Scene Classification, focuses on a novel approach based on LBP to capture the temporal dynamics between MFCC features. Some complementary spectral features such as spectral centroid (SC), spectral bandwidth (SBW) are utilized to further improve the ASC performance. Chapter 3 also presents the use of ensemble learning technique called D3C[16] to effectively classify the environmental sounds.

Chapter 4, Sound Event Detection in Real Life Audio, introduces the use of joint spectral and temporal features that are derived from NMF2D. An ensemble approach named RUSBoost[17] is utilized in the event detection process to alleviate class imbalance between the background and each event.

Finally, Chapter 5, Conclusion and Future Work, presents the concluding remarks about the merits and limitations of the proposed methods and directions for future improvements.

.

Chapter 2

Background

This Chapter is divided into three sections. The first section investigates acoustic descriptors that are commonly used for characterizing the environmental sounds. The second section summarizes classification techniques for ASC. Finally, the third section introduces the most common detection schemes for SED.

2.1 Acoustic Features

Acoustic features can describe the characteristics and properties of sound from various aspects. Good features should be discriminating among different classes and robust to noise. This section introduces several categories of audio features that have been employed in analyzing the characteristics of environmental sounds.

2.1.1 Time Domain Features

The following paragraphs describe the temporal features that have been used in ESR.

zero-crossing rate (ZCR)

The short time ZCR is defined as the number of times the waveform of a audio signal from positive to negative or back during a time interval. It measure of the average rate of sign changes, thus can estimate

the dominant frequency of the signal. It was used in an audio-based surveillance systems proposed in [18]. The short time ZCR can be expressed as

$$Z_n = \frac{1}{2N} \sum_{m=0}^{N-1} |\text{sgn}\{x[m]\} - \text{sgn}\{x[m-1]\}| w[n-m], \tag{2.1}$$

where

$$\text{sgn}\{x\} = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0, \end{cases} \tag{2.2}$$

$w[n]$ represents the windowing function of length N and n is the shift in number of samples at which we are interested in knowing the short time energy.

MPEG-7 audio waveform (AW)

This type of feature describes the audio waveform, representing the downsampled waveform envelope. It is defined as the maximum and minimum samples of a function within non-overlapping frames. The work in [19] used AW as a feature in environmental sound recognition.

Power based features

Temporal acoustic features based on signal power have also been employed in ESR. These features include short-time energy (STE), which is defined as the average energy of each signal frame, and is useful for detecting the transition between unvoiced and voice speech. The STE is given by

$$E_n = \sum_{m=0}^{N-1} (s[m].w[n-m])^2, \tag{2.3}$$

where $w[n]$ is the windowing function of finite duration N, and n is the the shift in the number of samples at which we would like to compute the STE. MPEG-7 temporal centroid has also been used [20]. It represents the time point in a signal where most energy is located on an average, and is defined as the

time average over the signal envelope using:

$$TC = \frac{N_{hop}}{F_s} \frac{\sum_{l=0}^{L-1}(lEnv(l))}{\sum_{l=0}^{L-1} Env(l)}, \tag{2.4}$$

where $Env(l)$ is the signal envelope given by

$$Env(l) = \sqrt{\frac{1}{N_w} \sum_{n=0}^{N_w-1} s^2(lN_{hop}+n)} \qquad (0 \leq l \leq L-1), \tag{2.5}$$

L is the total number of frames, N_w is the frame size, N_{hop} represents the hop size, and F_s is the sampling rate. The factor $\frac{N_{hop}}{F_s}$ represents the frame sampling rate.

2.1.2 Frequency Domain Features

Spectral features are another set of acoustic features reported in the literature of ESR [21] [19]. These frequency-domain descriptors can be obtained by converting the signal into frequency-domain using Fourier Transform. We summarize some of the spectral features based on the work in [22] [23]. Let us define $s_i[n]$ as the ith frame of an audio signal and $S_i[f]$ as the spectrum of this frame. Then, we divide the spectrum $S_i[f]$ into M subbands that are non-overlapping. The frequency range of each subband is from l_b to u_b.

Spectral centroid (SC)

SC is defined as the weighted average frequency for a given subband, which is a good measure of the center of gravity in each subband. It is given by:

$$\mathrm{SC_{i,b}} = \frac{\sum_{f=l_b}^{u_b} f|S_i[f]|^2}{\sum_{f=l_b}^{u_b} |S_i[f]|^2}. \tag{2.6}$$

spectral bandwidth (SBW)

SBW is defined as the weighted average distance from frequency to spectral centroid (SC) for a subband. This feature represents the relative spread of each subband. It can be computed by:

$$\mathrm{SBW_{i,b}} = \frac{\sum_{f=l_b}^{u_b} (f - SC_{i,b})^2 \ |S_i[f]|^2}{\sum_{f=l_b}^{u_b} |S_i[f]|^2}. \tag{2.7}$$

spectral band energy (SBE)

SBE represents the normalized energy of each subband. The SBE shows the energy distribution for a environmental sound, and therefore it tells us what the dominant frequency range is. It is given by:

$$\mathrm{SBE_{i,b}} = \frac{\sum_{f=l_b}^{u_b} |S_i[f]|^2}{\sum_f |S_i[f]|^2}. \tag{2.8}$$

spectral flatness (SF)

This feature describes uniformity in the frequency distribution in the spectrum. Therefore, noise-like sounds tend to have high value of SF. We can compute it using:

$$\mathrm{SF_{i,b}} = \frac{[\ \prod_{f=l_b}^{u_b} |S_i[f]|^2\ ]^{1/(u_b-l_b+1)}}{1/(u_b-l_b+1)\sum_{f=l_b}^{u_b} |S_i[f]|^2}. \tag{2.9}$$

spectral crest factor (SCF)

Contrary to SF measure, SCF provides a measure for identifying the peak of power spectrum in each subband. Different from spectral flatness, noise-like sounds would have lower SCF. It is given by:

$$\mathrm{SCF_{i,b}} = \frac{\max|S_i[f]|^2}{1/(u_b-l_b+1)\sum_{f=l_b}^{u_b} |S_i[f]|^2}. \tag{2.10}$$

Shannon entropy (SE)

SE is a measure that detects the randomness of each subband. We can calculate it by:

$$\mathrm{SE_{i,b}} = -\sum_{f=l_b}^{u_b} \left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right| \cdot \log_2 \left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right|. \tag{2.11}$$

Renyi entropy (RE)

RE is another measure that describes randomness of each subband. It is defined by:

$$\mathrm{RE_{i,b}} = \frac{1}{1-\alpha}\log_2 \left(\sum_{f=l_b}^{u_b} \left|\frac{S_i[f]}{\sum_{f=l_b}^{u_b} S_i[f]}\right|^{\alpha}\right). \tag{2.12}$$

2.1.3 Cepstral Domain Features

Cepstral domain features provide a compact representations of the signal spectrum by approximating its logarithmic magnitude.

Perceptual filter bank based features

Frequency domain features can be further analyzed using filter banks that mimic the characteristics of the human auditory system. Clarkson *et al.* [24] used Mel-scaled filter-bank coefficients (MFCs) to distinguish speech and non-speech environmental sound. By computing the discrete DCT of the logarithm of MFCs, MFCC that capture the spectral envelope of a sound are obtained. Although MFCC are primarily designed for speech processing, they still have been a popular feature of choice in the field of acoustic scene analysis (ASA). Aucouturier *et al.* [25] analyzed the bag-of-frames (BOF) approach to audio pattern recognition, and concluded that the MFCC+GMM approach is sufficient for recognizing urban soundscapes but not for polyphonic music.

Sawhney and Maes [26] used Gammatone filters to approximate the filtering done by the human cochlea. Gammatone filters were originally designed to mimic the human auditory spectral response, base on the ability to approximate the impulse response, magnitude response and filter bandwidth [27].

linear prediction cepstrum coefficients (LPCC)

This feature provide a robust and compact representation of the audio signal, and is developed for automatic speech recognition. It can be derived from computing the inverse Fourier transform of the logarithmic magnitude of the linear prediction spectral complex envelope [28]. It has also been found to be useful for ESR [29].

2.1.4 Spatial Features

This class of features are extracted from the different channels of the audio signal to identify the overlapping sound events. Nogueira *et al.* [30] utilized the interaural time difference (ITD), which measures the relative delay occurring between the left and right channels, and provides a cue to the direction the sound source; the interaural level difference (ILD), which the difference in loudness and frequency distribution between the two channels, in their proposed system. Adavanne *et al.* [31] employed time

difference of arrival (TDOA) features to help identify the overlapping sound events.

2.1.5 Time-frequency Features

Environmental sounds are non-stationary audio signals that have time varying characteristics. Time-frequency features can effectively represent these characteristics, and therefore are employed in a large amount of feature extraction methods for ESA. In the following paragraphs, we will discuss features derived from commonly used time-frequency analysis methods.

Spectrogram based methods

There has also been some prior work on extracting various image processing features from the spectrogram of an audio clip. Kobayashi and Ye [32] treated the spectrogram of an audio clip as an image and proposed some effective and robust LBP-based features by incorporating the local statistics in the spectrogram and using L_2-Hellinger normalization technique. Similarly, the work in [33] also used the LBP descriptors to capture the distribution of audio structure from the spectrogram.

By describing the spectrogram of an environmental sound as a linear combination of the basic functions, matrix factorization can be used to obtain a class of unsupervised learning features. The basic functions are considered to represent the signature of sound events, whilst the corresponding activation functions encode the contribution of the basic functions in time. Thus, global and local features can be obtained jointly. Hennequin *et al.* [34] and Ghoraani *et al.* [35] used non-negative matrix factorization (NMF), and Benetos *et al.* [36] employed probabilistic latent component analysis (PLCA) in their proposed algorithms .

Wavelet based features

The wavelet transform is similar to the Fourier transform (FT) except that the base function used. FT decomposes the signal into sines and cosine, while wavelet transform (WT) uses wavelets of different scales and positions. Compared to STFT, WT is more effective in representing non-periodic non-stationary signals that have discontinuities and sharp peaks [37]. Cowling and Sitte[38] applied fast wavelet transform (FWT) and continuous wavelet transform (CWT) to produce a TFR of environmental sounds. They concluded CWT is more suitable for recognition task, while FWT can effectively encode and decode signals.

Matching pursuit (MP) based methods

MP is another type of signal decomposition methods that represents signals using a finite dictionary of [39]. This algorithm sparsely decomposes a signal by selecting the optimal subset of atoms from a given dictionary. The basis functions constituting the dictionary can include wavelet functions, Gabor functions. At each step, the selection of optimal atoms is determined by maximizing the energy removed from the residual signal. This allows obtaining a reasonable approximation of the signal with a few basis functions, which provides an interpretation of the signal structure.

Umapathy *et al.* [40] used signal decomposition parameter based on octave to generate a set of features over several subbands within the auditory range. Chu *et al.* [29] utilized MP to select a small set of time-frequency features which were adopted to supply MFCC features, demonstrating that these joint features coupled with MFCC improved the performance of environmental sound recognition. The performance of their system was comparable to humans. The approach proposed by Ghoraani *et al.* [35] also demonstrated the effectiveness of MP algorithm. Their framework constructed time-frequency matrix (TFM) of an audio clip by using matching pursuit time-frequency distribution (MP-TFD) technique. Then the TFM were described as a linear combination of elementary functions by applying NMF technique. Besides, Schroeder *et al.* [41] investigated the use of Gabor filter-bank features that capture spectral, temporal and joint spectro-temporal modulation patterns of the sound.

Constant Q transform based methods

Rakotomamonjy and Gasso [42] investigated the use of histogram of gradients (HOG) for ASC. Their method comprises the following steps [11]. First, the audio signals in the dataset are processed using a constant-Q transform, which returns frequency representations with logarithmically spaced frequency bands. Next, they resized all constant-Q representations to a 512*512-pixel gray-scale image by using a bicubic interpolation and applied mean filtering on it. Finally, the features are extracted from the images by computing the matrix of local histograms of gradients. This is obtained by split images into non-overlapping cells, defining a set of spatial orientation directions, counting the occurrence of gradient orientations, and normalizing each cell histogram. Further these HOG features are locally pooled to gain more invariance and robustness.

2.2 Classification Methods

Once the features are extracted from the audio signal, the next stage of an ASC system generally comprises using a machine leaning algorithm to categorize the features vectors. In general, there are two types of models, namely generative models and discriminative models.

2.2.1 Generative Methods

These models are referred as generative models because of their capability of statistically generating a feature sequence. By assuming that feature vectors can be generated from one of a set of underlying distributions, generative models are used to learn the distribution of the extracted features. In the training phase, the parameters of the distributions are estimated using the statistics of the training data. In the test stage, a decision criterion is used to determine the most likely model that generated the test data.

By computing the basic statistical properties of the distribution of feature vectors, we can get one class centroid for each category. The same statistic can be computed for each unlabeled sample that we assume that it is generated according to the distribution with the closest centroid and assign this sample to the corresponding category.

Gaussian mixture models

Gaussian mixture models (GMMs) are always the model of choice for acoustic and speech modeling, as they are quite flexible and computationally efficient to train. By increasing the number of mixture components of the model, GMMs can also approximate complex density. Suppose $\mathcal{N}(x, \mu_k)$ refers to a normal distribution with mean μ and covariance matrix μ_k, then feature vector x extracted from the training data is assumed to be generated by the following distribution:

$$x \sim \prod_{k=1}^{K} w_k \mathcal{N}(\mu_k, \Sigma_k) \tag{2.13}$$

where K is the number of Gaussian mixtures, and w_i represents the probability that this observation is generated from the ith component. In this case, the only parameter need to be set is the number of Gaussian mixtures K, which affects the model accuracy and overfitting. On the one hand, the model

need sufficient number of components because each acoustic scene contains events with various spectral properties. On the other hand, as the number of components becomes too large, the model tends to lose the generalization capabilities. Thus, when confronted with an unlabeled sound, it may fail to recognize it.

Once the parameters of each GMMs have been inferred from the training data, features can be extracted from an unlabeled sound. These feature vectors would be normalized using the same mean and standard deviation values as the training data, and a decision criterion is employed to determine which model is statistically most likely to generate the observed features, hence classifying the sound. When using GMMs, the ordering of the sequence of features do not affect the parameters of the model, thus, the classification results.

Hidden Markov models

The hidden Markov models (HMMs) have also been used in several ASC systems. Extending the modeling ability of GMMs, this approach account for the temporal evolution of events within complex acoustic scene. This method specifically works for the scenes where event would occur sequentially. For instance, for an acoustic scene recorded in the subway station, we would hear an alert sound before the door closing sound, followed by the sound of train moving. This method firstly uses GMMs to model the extracted feature vectors, then temporal information of the event occurring order would be encoded as the transition probability.

The system being modeled with HMM is assumed to be a Markov process with a set of hidden states, where for each state the output probability distribution is modelled using a GMM, and the transitions between different states are determined by the corresponding probabilities in the transition matrix. Given a sequence of feature vectors, our goal is to calculate the most likely states that could generate these observations . Let us define y_t as the observation at a given time instance, K is the the number of hidden states $q_t \in \{1..., K\}$. In the training stage, we need to estimate a set of parameters θ, including the initial state $\pi(i) = P(q_1 = i)$, the transition matrix, $A(i, j) = P(q_t = j|q_{t-1} = i)$, as well as the observation probability distribution $P(y_t|q_t)$. These can be obtained using Baum-Welch algorithm [43] to maximise the likelihood of the training data, Y :

$$\theta^{k+1} = \arg \max_{\theta} P(Y|\theta^k) \tag{2.14}$$

In the test stage, we need to find the most likely state sequences that would have generated the observed sequences, which is called Viterbi decoding. The most probable state sequence q_{best} is then defined as:

$$\begin{aligned} q_{best} &= \arg \max_{q} P(Y, q|\theta) \\ &= \arg \max_{q} P(Y|q, \theta)P(q|\theta) \end{aligned} \tag{2.15}$$

I-vector

The ASC system proposed by Elizalde *et al.* [44] is based on the computation of the I-vector. I-vector is originally developed for speech processing to address the problem of speaker verification, and it is based on modeling a sequence of features using GMMs. In the context of ASC, the I-vector is considered as a function of the parameters of the GMMs learned from the MFCC features. It leads to another representation summarizing the properties of a soundscape.

2.2.2 Discriminative Methods

When using a discriminative classifier, we do not consider the features as being generated by a underlying distribution. Instead, they are assumed to occupy a class-specific region in the feature space. Given a training set, an discriminative algorithm tries to find a decision boundary that separates the different classes. Then, to classify a new sample, it checks on which side of the decision boundary it falls, and makes its prediction accordingly. Discriminative and generative models can be combined together. For instance, the parameters of the generative models learned from training data can be used as features and then employ a discriminative classifier to learn separating boundaries. In other words, discriminative classifiers can be used to determine classification criteria from either the original feature vectors or the parameters of their statistical models.

Support vector machine

For ASC, one of the most popularly used discriminative classifiers is the support vector machine (SVM). According to a maximum-margin criterion, an SVM determines a set of hyperplanes that can optimally separate features from different classes in the training data. While a SVM can only discriminate between

two classes, multiple SVMs can be combined to determine a decision criterion that works for multi-class problem. There are two combination schemes. In the one-versus-all approach, a SVM is trained for each class to discriminate between data belonging to this class and data from the remaining classes. However, in the one-versus-one approach, for every possible combinations between two classes, a SVM is trained. In both cases, the decision criterion determines the class of an unlabeled sample by comparing the distance between the data and the different separating hyperplanes learned by the SVMs.

2.2.3 Ensemble Learning Techniques

In the context of supervised classification, ensemble techniques can be applied to increase the classification accuracy by running multiple instances of a classifier with different parameters in parallel. The components of a classification algorithm can themselves be thought of as parameters subject to optimization. Thus, a further class of meta-algorithms deals with selecting from or combining multiple classifiers to improve the classification accuracy. The results from each classifier are finally combined into a global decision.

Majority vote and boosting

In this combining method, an unlabeled instance is classified to the class that obtains the highest number of votes. This method is the most basic ensemble method. Other more complicated methods include boosting techniques [45]. By learning several base learning from the different weighted examples, this method allows the weak learner to focus on correctly classifying the most highly weighted examples while strongly avoiding over-fitting. During testing stage, each of the base learner get a weighted vote proportional to their accuracy on the training data.

Bagging

This method creates ensembles by repeatedly randomly resampling the training data. The resulting bootstrap samples are then used to fit the models, and then the result is obtained from combining all of the resulting models using simple majority vote. This algorithm is designed to improve the stability and accuracy of machine learning algorithms, while avoid overfitting. Li *et al.* [46] employed a treebagger classifier to form a collection of decision trees. Different from the generative and discriminative models, a decision tree is a set of rules learned from analyzing features extracted from training signals. These

rules then lead to a classification output. In the method proposed in [46], the resulting weak learners are then combined to determine a category for each frame and, during the test phase, an overall category is assigned to each acoustic scene based on a majority vote.

2.2.4 Deep learning algorithms

Various deep learning algorithms have also been adopted for improving the performance of ASC sytem [47] due to their effectiveness in the recognition problem. The work in [48] applied several kinds of models including the deep neural networks (DNN), recurrent neural networks (RNN) and recurrent deep neural networks (RDNN). They found deep learning models outperformed the traditional models such as GMMs and SVM. In order to utilize the temporal information of the signal, Bae *et al.* [49] used convolutional neural networks (CNN) to learn the spectro-temporal locality from the spectrogram.

2.3 Detection Schemes

For SED, the detection module is concerned with finding the start and end points of each sound event, and segmenting it from the continuous audio stream. In general, approaches use one of two methods to deal with the detection problems: detection-and-classification, or detection-by-classification, where the latter combines detection and classification into a single pattern recognition problem.

2.3.1 Non-negative Matrix Factorization

NMF is one of the most popular techniques for detecting polyphonic sound events. It can be used in different stages of the SED system. For SED completely based on NMF, the activation function obtained from the test data served as the event predictor. In the training stage, the dictionary matrix W can be obtained from the training samples. In the testing stage, the activation matrix H of the test data can be derived by the learned dictionary matrix W. Then, the activation matrix of the test data can be further processed to produce the detected event. The work in [50] applied this method to overcome the need to assign separated components to sound event classes.

2.3.2 Detection and Classification

The detection task can be accomplished by two separate classifiers: one binary classifier for identifying whether events occur and the other for labeling the detected events. Furthermore, a median filter can be applied on the label sequences to ensure the minimum event gap is met[51]. Finally, an event hypothesis is excluded if its length is less than the minimum length required.

2.3.3 Statistical Method

This method is the same as the one used in ASC where GMMs are used to model the states of frame-wise features, and then HMMs can learn the distributions of the feature sequences given the state sequences. In the training stage, a binary classifier is set up for each event class. The class model is trained using the audio segments annotated as belonging to the modeled event class, and a negative model is trained using the rest of the audio. In the testing step, The decision is based on likelihood ratio between the positive and negative models for each individual class, with a sliding window of one second.

2.3.4 Regression Method

In this method [52], a binary segment wise classifier is used for event and background classification. Subsequently, the event segments are classified by a multi-class event classifier. Then, a regression forest is trained for each event category to estimate the event onset and offset. In addition, the event estimation is weighted by the classification probabilities obtained from the previous multi-class event classifier. Finally, detection thresholds are eventually applied to the estimated onset and offset to determine event start and end points.

2.4 Limitations

The previous section introduced current approaches for ESR in the literature. We find that features such as MFCCs or mel energies are widely used. While these features may provide a reasonably good representations, these traditional features do not capture the temporal information sufficiently. Since acoustic scenes can be characterized by certain sound events, and sound events may exhibit unique temporal patterns, for ASC, we need to represent the temporal dynamics of a signal in a global context.

However, for SED, features with local temporal information are needed to detect sound events. In addition, the overlapping sound events make it harder for the system to detect individual events from the mixture. In the classification phase, the best performance is always achieved by the late fusion of several models [53][54]. The performance of this fusion method mainly depends on the artificially selected models, and there is no specific criterion that can be followed when choosing from a range of classifiers. For the detection problem, the state-of-the-art methods are based on the deep learning models with the multi-label classification ability. Traditional classifiers such as SVM often fail to solve this problem due to the class imbalance between classes. Limitations of current methods can be summarized as: insufficient features describing the temporal information of the signal and insufficient use the ensemble learning techniques. These motivate the research in this thesis to find an alternative approach to address the problems faced in SER.

2.4.1 Temporal Information Extraction

While classic features such as MFCCs are commonly used for ESR, the existing temporal feature integration methods, such as taking simple statistics (i.e., mean and variance) of the feature vectors over time, does not fully capture the temporal evolution characteristics of the sound. Although modeling the temporal dynamics of the frame-based features can be conducted in HMMs [55], temporal distribution of environmental sounds over various time periods make it hard for HMM to be optimized. Therefore, finding other ways to capture the temporal information may be useful for improving the performance of ESR .

2.4.2 Ensemble of Models

As mentioned in the previous section, ensemble learning technique has been employed in ESR. However, ensemble methods found in the literature are only some simple and basic ones. For example, majority vote is commonly used to combine the output of several classifiers. In this case, the base classifiers are often picked up by experience, and we can only test the results by using different combinations. It is hard to tell whether the best combination has been found, thus we may not get the optimal results.

Ensemble learning is a machine learning technique that select a collection of hypotheses and combine their predictions. When combing multiple independent and diverse decisions each of which is at least

more accurate than random guessing, random errors cancel each other out, correct decisions are reinforced [56]. For ASC, ensemble techniques can be utilized to generate a set of base classifiers and combine them in order to produce higher classification accuracy. Ensemble-based methods that address problem of class imbalance include bagging, boosting, and hybrid-based approaches [57]. Sound event datasets that often suffer from the imbalanced class distribution, therefore these techniques can be used to improve the detection performance.

2.5 Summary

This chapter has given the current state-of-the-art of ESR. This was provided by first giving a review of the traditional approaches for feature extraction, with the acoustic features grouped into five groups. Then, a range of classification methods for ASC was reviewed, with the techniques grouped into three categories. Next, current detection schemes to SED were given. Finally, the limitations of current approaches were discussed. Together, this chapter motivates the work in this thesis to find novel ways of capturing the sound information. The next chapter introduces the proposed approach, where the idea is to extract features characterizing the temporal information and utilize ensemble learning method.

Chapter 3

Acoustic Scene Classification

In the previous Chapter, a review of current techniques for environmental sound recognition and their limitations were given. Although many of these methods are based on a frame-based frequency analysis of the audio signal, it is important to analyze the temporal structure of the data as representing their frequency characteristics alone is not enough for classification. In this chapter, the idea of capturing the temporal dynamics of MFCC using image processing technique is introduced. This is referred to here as spectrogram image processing, where features are extracted from the two-dimensional spectrogram image to jointly characterise the time-frequency sound information. This Chapter is organized as follows. Section 3.1 provides motivation behind the idea of extracting the temporal evolution by LBP. Based on this, a new approach for combining the temporal features is then proposed in Section 3.2 for ASC. Experiments are then conducted, and the results and discussion are given in Section 3.3 to evaluate the performance of the proposed method.

3.1 Motivation

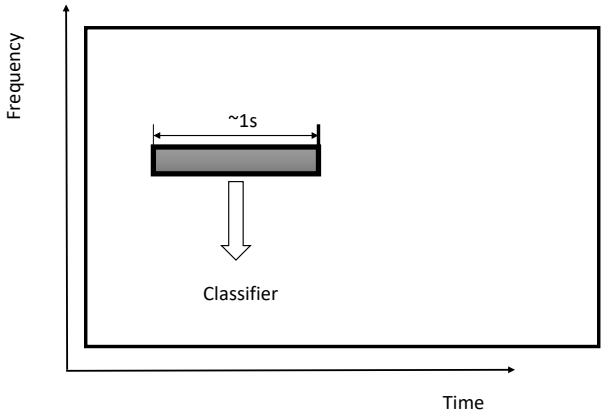
In the analysis of audio signals, much emphasis has been placed on the spectral information and the temporal properties of the signal usually have not been considered. In this case, it is assumed that the features in different frames are independent of each other. As acoustic signals may consist of various sound sources, their frequency characteristics alone are not distinctive enough to be used for classification, and the information on the temporal evolution of these features are also useful [58]. A basic approach

is to use the delta and delta-delta coefficients to capture the temporal transitions of the frame-based spectral features [59]. Besides, some other features are designed to characterize the evolution of these features over time.

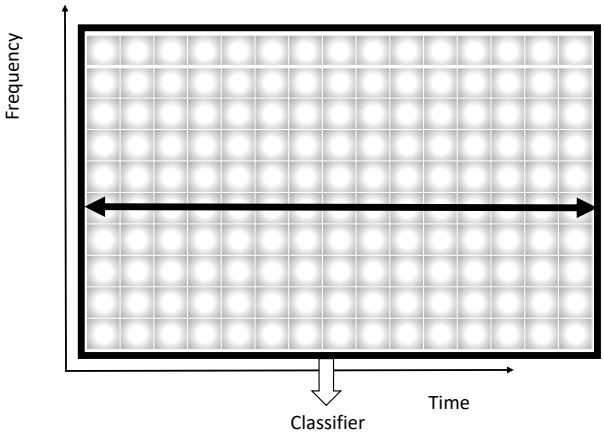
As demonstrated in Figure 3.1, the temporal information can be extracted across a range of different time and frequency scales, including capturing the joint spectral and temporal information in the feature sequences. The approach in [60] focused on extracting the temporal patterns from the frame-based features as in Figure 3.1a, where features are extracted over an one-second temporal window from each frequency subband. The temporal patterns of spectral energy were found to be able to classify phonemes with a reasonable accuracy.

Another type of approach for capturing the temporal dynamics in the signal (see Figure 3.1b) is to use temporal feature integration, which refers to the process of combining a sequence of feature vectors into a single segment-level feature vector while capturing the relevant temporal information in the time series of these features [61]. There are two differing integration processes: early and late integration [58]. Early integration operates at the feature level, and aims at combining all the feature vectors extracted over short-time windows into a single vector. Late temporal integration, on the other hand, works at the decision level and does not capture information about the temporal dynamics [58] [62].

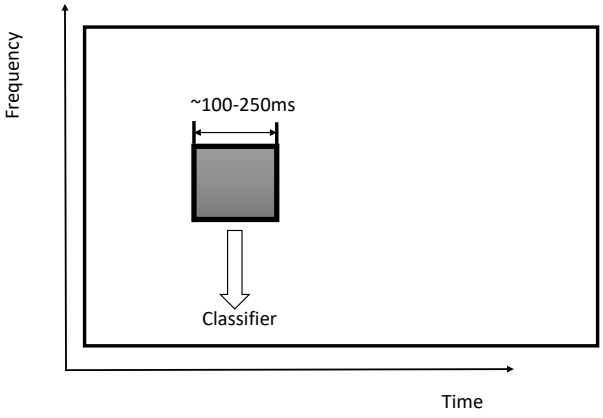
Often, simple statistics such as the mean/variance are used for integrating temporal features. However, these simple statistics do not consider the temporal dynamics among successive feature sequences. Meng *et al.* [61] proposed a multivariate autoregressive feature model to capture the temporal dynamics for music genre classification. Joder *et al.* [58] introduced a number of early and late temporal integration methods, and discussed their impact on the performances of musical instrument recognition systems. They found that the combination of early and late integration brought improvements of the recognition rate. An example of the late temporal integration is the use of HMMs to model the temporal dynamics. We have discussed two temporal paradigms in audio signal processing, and the joint spectro-temporal features will be discussed in the next chapter. Although many of the existing works in the context of ASC are based on MFCC and their derivatives, only a limited number of them focus on the integration of these temporal features. Often, simple statistics such as the mean and variance are used for integrating temporal features. In the recent work [63], features in the field of chaos theory are introduced to describe the temporal evolution of MFCCs. Recurrence Quantification Analysis (RQA) is used to capture the auto-correlation through the similarity matrix of frame sequences. Nevertheless, RQA



(a) Temporal patterns



(b) Temporal integration



(c) Joint spectro-temporal

Figure 3.1: Temporal paradigms for characterizing the time-frequency representations

operates on the similarity matrix of MFCCs, instead of catching the temporal evolution directly from MFCCs. The work in [64] reports a hybrid GMM-HMM system that combines both mean modulation statistics tracked obtained from the GMM model with the temporal trajectories tracked by the HMM model. It shows that temporal dynamics of modulation features do provide complimentary information in addition to their mean statistics.

In this chapter, we concentrate on capturing temporal dynamics using texture features from image processing techniques. We apply local binary pattern (LBP) [65] to MFCCs that are extracted over frames, in an attempt to capture some additional information carried by temporal evolution of these characteristics.

3.2 Combining Temporal Features for Acoustic Scene Classification

Inspired by the idea of capturing the temporal dynamics from the signal for ASC, a novel feature extraction method is now presented. The motivation stems from the fact that LBP can be used to encode the pixel changes across different ranges, the overall LBP features of the image can then be summarized using the histogram. Thus LBP is utilized to capture the missing information about the temporal dynamics caused by the integration process.

The use of LBP features in ASC has already been proposed in other works [32] [33]. By applying LBP analysis to spectrograms, the work in [33] aims to capture the distribution of audio structure, whilst the work in [32] intents to extract the geometrical characteristics on the spectrogram. On the one hand, the successful applications of LBP to spectrogram demonstrate its effectiveness to characterize TFR. On the other hand, the LBP, which is obtained by the comparison of outer values to the center value, seems to fit our goal of characterizing the evolution of MFCCs over time. Different than the two previous LBP works for ASC, we apply LBP to frame level MFCCs in order to extract features complementary to MFCCs, rather than extracting features from the audio.

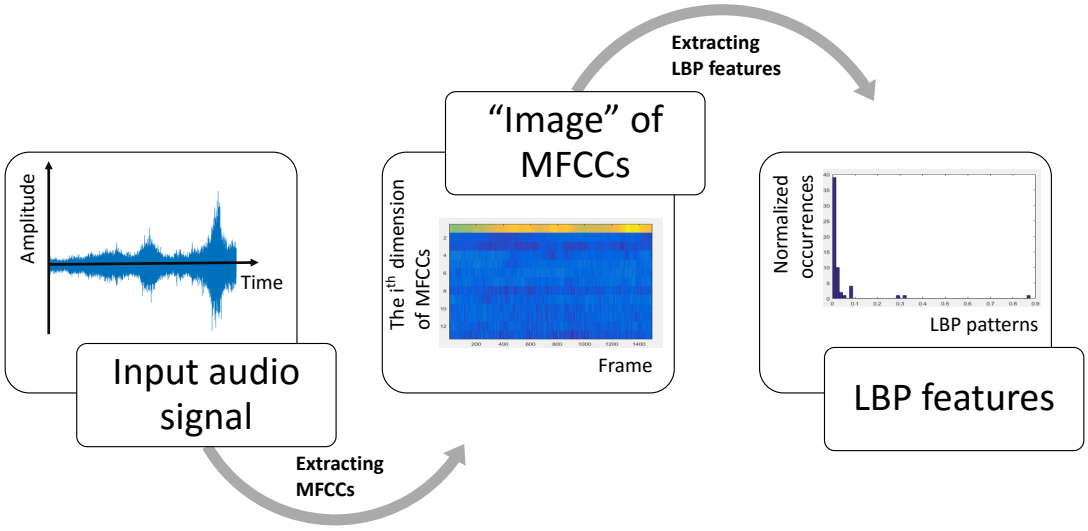
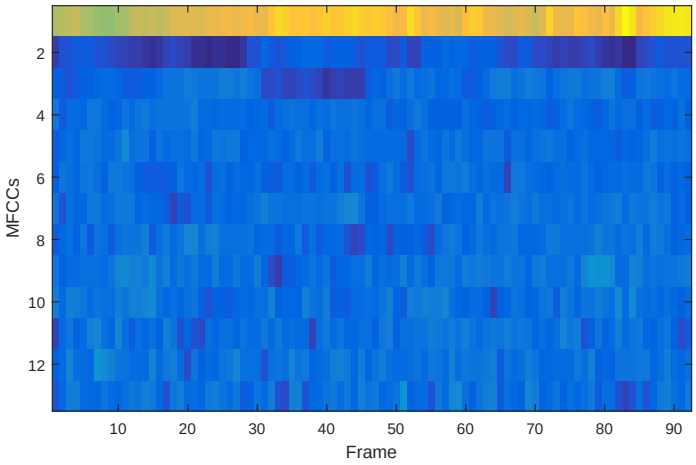


Figure 3.2: Using LBP descriptor for extracting the temporal evolution of frame-level MFCC features, presented in [66] ©2017 IEEE

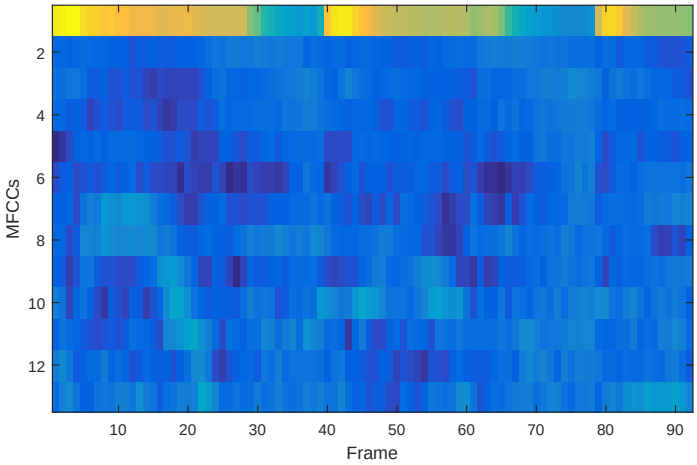
3.2.1 Overview

In this section, we will introduce the features and classifier used in the proposed system. As shown in Figure 3.2, the LBP features can be extracted using the following steps. At first, MFCC features are extracted from frames of audio signals. Before these features are integrated into a single vector, LBP is utilized to capture the temporal dynamics among the MFCC sequences. The combinations of LBP based features and MFCC are then fed to the classifier. In the classification phase, we adopted an ensemble classifier called D3C [16], which is a combination of the ensemble pruning based on k-means clustering and dynamic selection and circulating combination.

As shown in Figure3.3, the MFCC of audio segments from different acoustic scene tend to have diverse temporal transition characteristics. Then, LBP is applied to the MFCC sequences to extract the temporal patterns. The squared error is visualized in Figure 3.4 the to compare the difference between the same scene (Office versus Office) and distinction between different scene (Office versus Home). It can be seen that the squared error is smaller when audio segments are from the same scene.



(a) MFCC of an audio segment from the scene home



(b) MFCC of an audio segment from the scene office

Figure 3.3: MFCC of different acoustic scenes

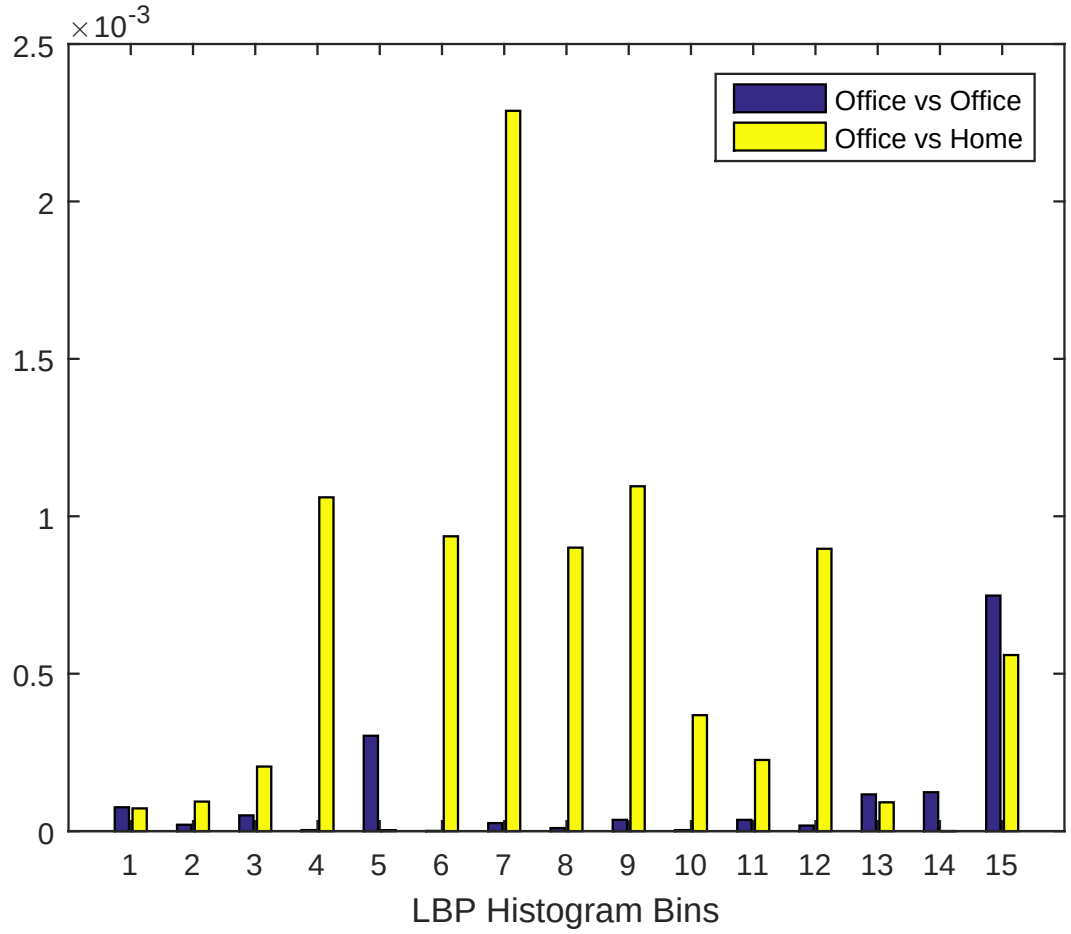


Figure 3.4: Differences between LBP Histograms obtained from MFCC of different acoustic scenes.

3.2.2 Feature Extraction

MFCC

MFCC features are one of the the most commonly used features in many audio signal processing tasks such as speaker recognition and music genre classification. MFCC are based on human hearing perceptions and the known variation of the human ears critical bandwidth with frequency. The perception of these frequencies by human brain is modeled though the use of the Mel lter bank. It has two types of filter which are spaced linearly at low frequency below 1000 Hz and logarithmically spaced above 1000Hz.

As shown in Figure3.5, MFCC are derived using the following steps [67]:

1. Segment the signal into short frames.
2. Take the Fourier transform of each frame.
3. Apply the Mel lter bank and sum the energy in each lter.
4. Take the logarithm of the energies.
5. Take the Discrete Cosine Transform (DCT) of the Mel log energies.
6. Keep the first n amplitudes of the resulting spectrum.

Sub-band MFCC [68] are a little different from the standard MFCC. They are computed using frequency banks that are only distributed between the specified lower and upper frequency bounds of each band.

The delta and delta-delta coefficients are also calculated to determine the rate of change and how fast these changes occur. These features are the most simple and direct trajectories of the MFCC coefficients over time. They are essentially the rst and second derivatives of the MFCCs and can be thought of as speed and acceleration of the changes. The delta coecients are calculated using the MFCCs and the following formula:

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n=1}^{N} n^2} \tag{3.1}$$

where d_t refers to the delta coefficient of frame t. It is computed in terms of the static coefficients c_{t-N} to c_{t+N}. Delta-delta (Acceleration) coefficients are calculated using the same method, but they are calculated from the deltas,rather than the static coefficients.

LBP Features

The LBP features can be computed through the following steps. Firstly, we extract Mel Frequency Cepstral Coefficients (MFCCs) from an environmental sound with time window of the length 40 ms (with 50% hop size). Then, the frame-level MFCC representations are viewed as a 2-D image on which we extract the LBP features as described in the following paragraphs, resulting a normalized histogram vector that contains supplemental information on the temporal dynamics. A novel feature vector is produced by concatenating LBP features to the averaging MFCCs.

The LBP algorithm was originally described by Ojala *et al.* [69] for texture analysis. Due to its invariance to monotonic gray-scale changes and low computational cost, LBP is widely used in image recognition and can successfully characterize the local patterns with robustness to fluctuations in pixels.

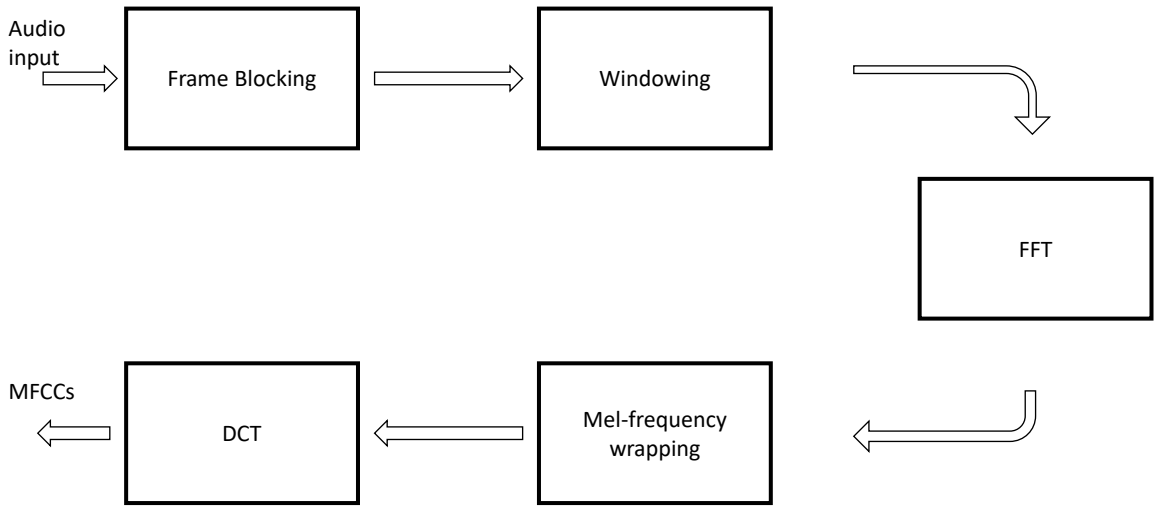


Figure 3.5: Block diagram of computing MFCC

The following description of the LBP operator is based on work in [70].

As shown in Figure 3.6, texture at g_c is modeled using a local neighborhood of radius R, which is sampled at P (8 in the example) points.

Let us define texture T in a local circular neighborhood denoted by (P, R) of a gray scale image as the joint distribution of the gray levels of $P(P > 1)$sampling points, given by

$$T = t(g_c, g_0, \ldots, g_{p-1}), \tag{3.2}$$

where g_c corresponds to the gray value of the center pixel (x, y) of the local patch and $g_p(p = 0, \ldots, P-1)$ refer to the gray values of P equally spaced pixels $(x_p, y_p) = (x + R\cos(2\pi p/P), y - R\sin(2\pi p/P)))$ on a circle of radius $R(R > 0)$ that form a circularly symmetric neighbor set. Figure 3.7 illustrates LBP operators with multiscale neighborhoods.

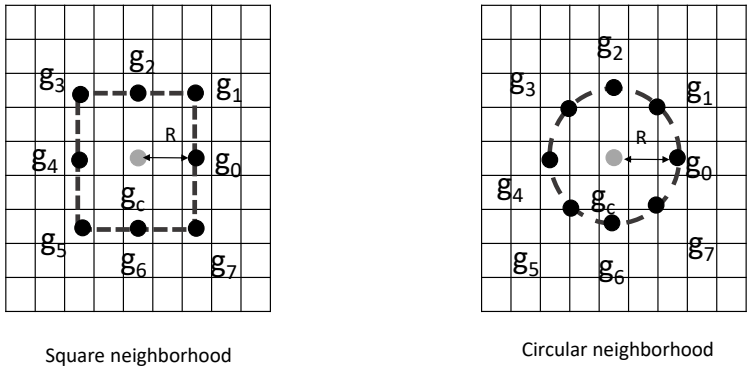


Figure 3.6: Summarizing the local structure in an image by LBP.

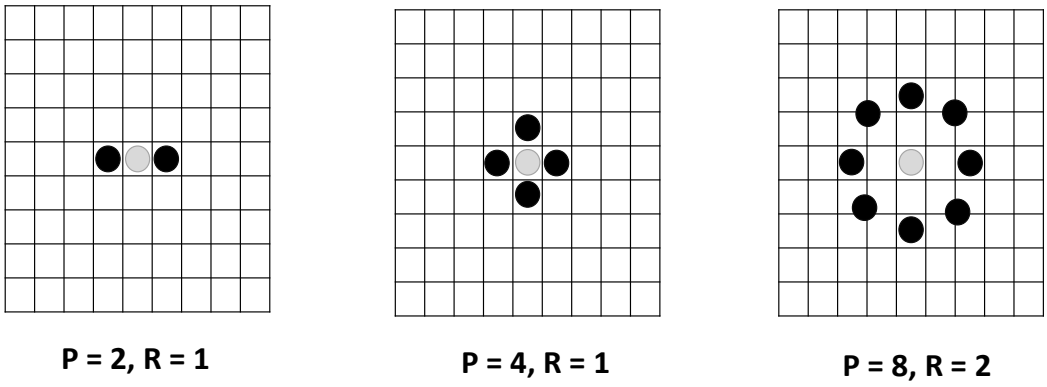


Figure 3.7: Multi-scale LBP.

LBP with gray-scale invariance

Firstly, we subtract the gray value of the center pixel g_c from the gray values of the circularly symmetric neighborhood $g_p(p = 0, \ldots, P-1)$. This gives:

$$T = t(g_c, g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c). \tag{3.3}$$

Then, we assume that differences $g_p - g_c$ are independent of g_c, allowing us to factorize (3.3):

$$T \approx t(g_c)t(g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c). \tag{3.4}$$

Since the distribution $t(g_c)$ in (3.4) describes the overall luminance of the image, which is unrelated to local image texture and not able to provide useful information. We can simplify (3.4), giving:

$$T \approx t(g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c). \tag{3.5}$$

By far, we get a highly discriminative texture operator that records the occurrences of various patterns in the neighborhood of each pixel in a histogram. Further, invariance with respect to the scaling of the gray scale can be achieved by considering just the signs of the differences instead of their exact values:

$$T \approx t(s(g_0 - g_c), s(g_1 - g_c), \ldots, s(g_{P-1} - g_c)), \tag{3.6}$$

where $s(x)$ is the thresholding function given by:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3.7}$$

In this way, the texture information of the center pixel (x, y) can be represented by a unique LBP number:

$$LBP_{P,R}(x, y) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p. \tag{3.8}$$

An example of applying LBP operator on a 3 x 3 neighborhood is given in Figure 3.8.

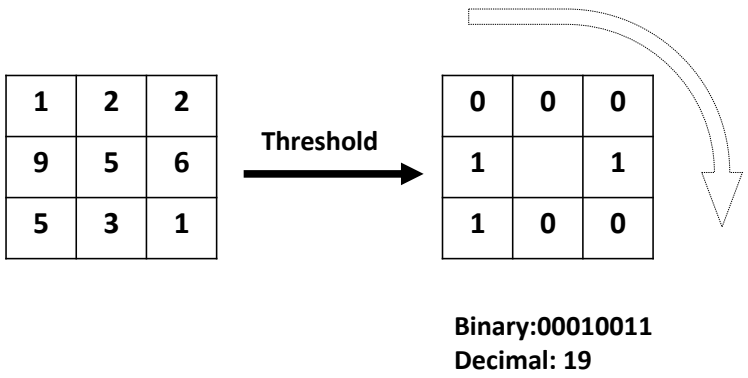


Figure 3.8: An example of applying LBP operator on a 3 x 3 neighborhood.

The uniform LBP (LBP^{u2})

A LBP descriptor is referred as a uniform pattern when the number of bitwise 0/1 changes is less or equal to 2. For example, patterns 11111111 (0 transition), 11100011 (2 transitions) are uniform, while patterns like 01000100 (4 transitions), 10010101 (6 transitions) are non-uniform ones. In practice, the occurrences of each uniform LBP are recorded in a unique histogram bin, while all non-uniform LBP are classified into one category and are cast into the same bin in the histogram. Although the number of output values produced by the uniform LBP is reduced from 2^P to $P*(P-1)+2$, we will not lose much important information. It was concluded by Ojala *et al.* [70] that vast majority of all local patterns can be categorized to be uniform.

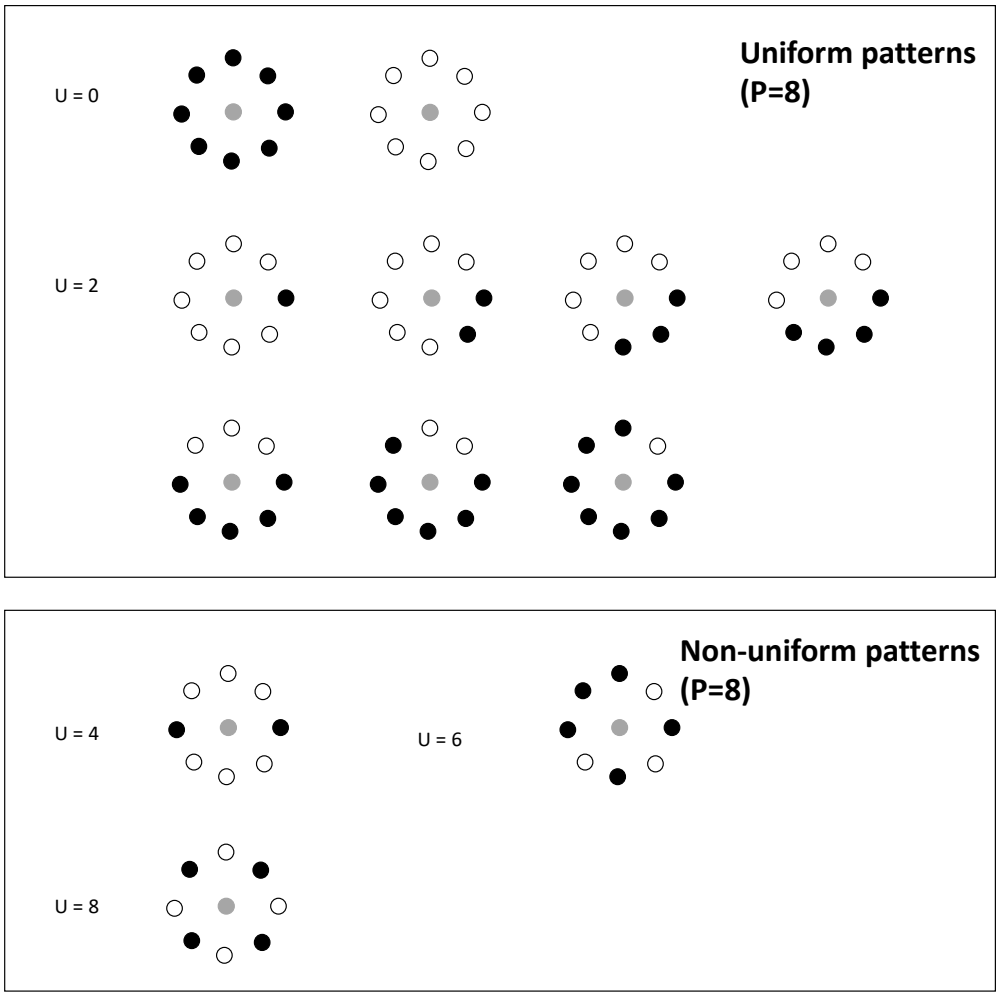


Figure 3.9: Examples of uniform and non-uniform LBP.

The uniform LBP with rotation invariance ($\mathbf{LBP}^{riu2}$)

Further, the rotation invariant uniform LBP is introduced to avoid changes of the value of LBP descriptor when the image is rotated. Those LBP features that can be characterized by the same value after a rotation operation would be cast into the same bin in the histogram, giving:

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & if\ U(LBP_{P,R}) \leq 2 \\ P+1 & otherwise, \end{cases} \tag{3.9}$$

where $U(LBP_{P,R})$ corresponds to the number of bitwise transitions.

Computational complexity of LBP features

For each input audio sound, we compute the LBP features with a P-point neighborhood on N-dimensional MFCC features of all k frames. The cost of computing all LBP descriptors would be $O(kPN)$. Therefore, the computational complexity is a linear function of the total length of the sound with constant MFCC dimensions and sampling points, and these features can be computed in real time.

Complementary Spectral Features

As we have introduced in Chapter2, there have also been a variety of spectral features to characterize the information embedded in the spectrum. Apart from the MFCC and LBP based features, spectral features including spectral centroid, spectral bandwidth, spectral band energy, spectral flatness, spectral crest factor, Shannon entropy,and Renyi entropy are added to represent the spectral characteristics.

3.2.3 Classification using D3C Ensemble Classifier

In the classification phase, we choose a ensemble classifier called D3C [16], which is a combination of the ensemble pruning based on k-means clustering and dynamic selection and circulating combination, since it optimizes the classification results by combining the outputs of a set of base classifiers.

For classification tasks, a typical ensemble method generally includes the following steps:

1. Ensemble generation - A number of base classifiers are generated according to a chosen learning procedure.

2. Ensemble pruning - A number of base classifiers are filtered out based on various mathematical procedures to improve the diversity of the classifiers as well as the overall ensemble accuracy.

3. Ensemble Combination - The outputs of the filtered classifier are

Based on the "overproduce and choose" strategy introduced by [71], D3C is a hybrid model that employs two types of selective ensemble techniques: ensemble pruning based on k-means clustering, and dynamic selection by circulating combination. It has been proved to exhibit competitive performance against other methods [16]. This hybrid model can be built through the following steps. At first, a

number of candidate classifiers selected from the base classifiers are trained. Then, some redundant classifiers are eliminated using ensemble pruning based on k-means clustering and the framework of dynamic selection and circulating combination is applied to choose the classifiers that have a high degree of diversity. Finally, the output results of classifiers are combined according to the combination rule to predict the label of an audio clip. The combination rules includes majority voting, average probability, maximum probability and so on.

In the following experiments, SVM is also used in order to be compared with the D3C classifier.

3.3 Experiments

In this section, experiments are carried out to find the parameters that give the best result, and compare the performance of the proposed method with the baseline method for ASC. In addition, several methods that are also inspired from capturing temporal dynamics of features are implemented, to provide a comparison between other methods more similar to the LBP.

3.3.1 Dataset

In order to evaluate the proposed method, we use the TUT Acoustic scenes 2016 development dataset [72], which are provided with the ASC task of DCASE 2016 challenge. This database consists of 1170 audio clips of 30-s duration each with a sampling rate of 44.1 kHz and a resolution of 24 bits/sample. All the recordings are available in WAV format and they are belonging to 15 different environment types. For all acoustic scenes, the recordings were captured each in a different location: different streets, different parks, different homes. For each recording location, 3-5 minute long audio recording was captured. The original recordings were then split into 30-second segments, and 78 segments were included in each scene. As shown in Figure 3.10 the fifteen acoustic scenes included were: Bus, Cafe / Restaurant, Car, City center, Forest path, Grocery store, Home, Lakeside beach, Library, Metro station, Office, Residential area, Train, Tram, and Urban park.

]

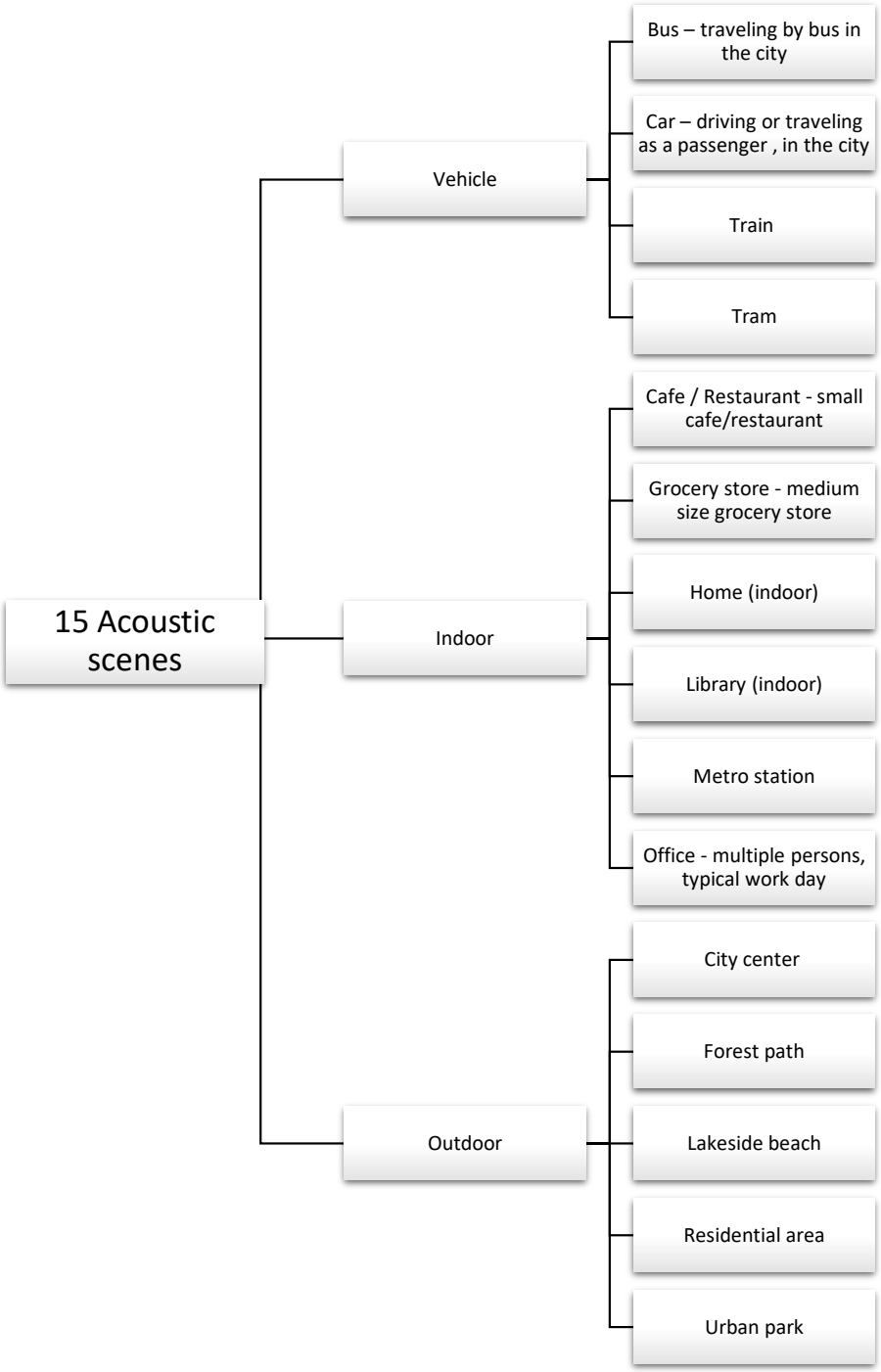


Figure 3.10: The organization of all audio signals in the TUT Acoustic scenes 2016 dataset.

3.3.2 Experimental setup

Cross-validation setup

The dataset was further partitioned into four folds of training and testing sets to be used for cross-validation. This process is illustrated in Figure 3.11. 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. This is a very important detail that is sometimes neglected, and failing to consider it results in overestimating the system performance, as the classification systems are capable of learning the locations related acoustic conditions instead of the intended general audio scene properties. This phenomenon is similar to the album effect encountered in music information retrievawhere accuracy improves when systems are trained and evaluated using music from the same ablum. Thus this performance characteristic is usually accounted for when setting up experiments [73].

LBP Evaluation Methods

Since LBP features are determined by a set of parameters, we conduct the following experiments to investigate some factors in generating LBP features:

1. Uniform vs. Rotation invariant uniform LBP
2. Neighborhood size (P, R)
3. Normalization method

Complementary Spectral Features

Apart from the MFCC and LBP based features, we combine some spectral features including spectral centroid, spectral bandwidth, spectral band energy, spectral flatness, spectral crest factor, Shannon entropy,and Renyi entropy with the MFCC and LBP features to see whether they can improve the classification result.

Baseline Methods

The baseline system [74] provided with the dataset uses 60-dimensional MFCC features and Gaussian mixture models (GMMs). The MFCC features consist of 20-dimensional MFCCs, delta coefficients and acceleration coefficients. A GMM with 32 components was used to examine the spectral change over

]

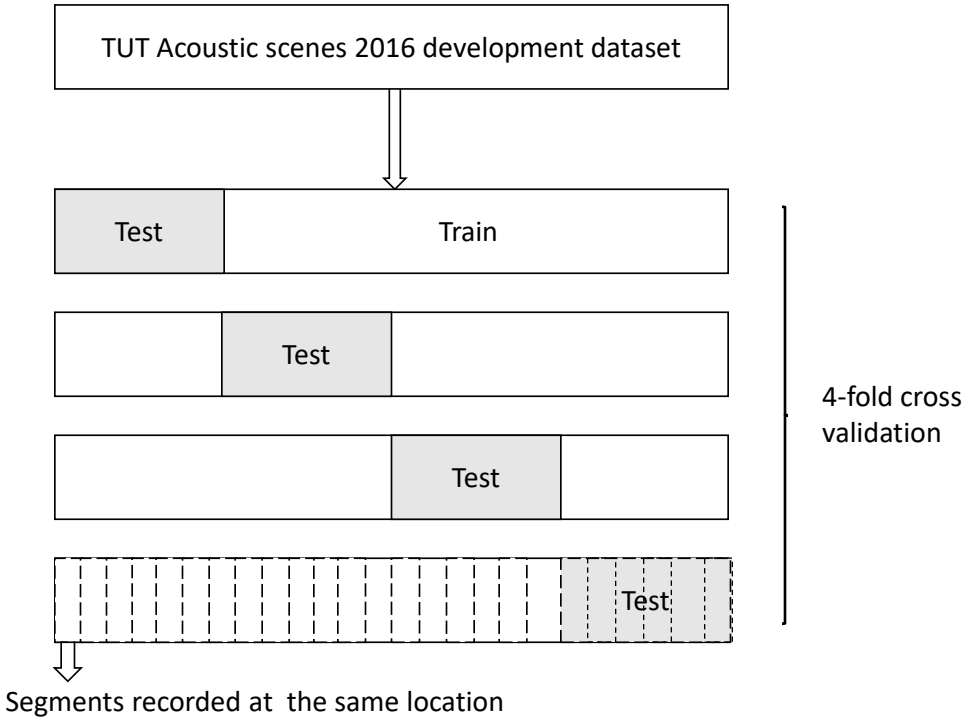


Figure 3.11: The organization of all audio signals in the TUT Acoustic scenes 2016 dataset.

time and trained for each class using the expectation maximization (EM) algorithm. In the testing phase, maximum likelihood decision was used to determine which class each sound belongs to. An overall classification accuracy of 72.5% was reported based on 4-fold cross validations. However, this model tend to perform on the extremes, as it performed well for office while producing extremely poor results for classes such as park and train.

In addition to the baseline MFCC+GMM method provided with the dataset, the RQA [63] method that is also inspired from temporal dynamics is also implemented to provide a more complete comparison with LBP based features.

3.3.3 Results and Discussion

The performance of the system is evaluated using classification accuracy: the number of correctly classified segments among the total number of segments. Each segment is considered an independent test sample.

Uniform vs. Rotation invariant uniform LBP

By applying LBP to capture the temporal evolution of MFCC, we actually encode the changes of each dimension of MFCC between frames. Since the frame sequence characterizing the same scene can appear in different order, rotation invariant LBP seems to be suitable for our task.

We first investigated the effect of two different kinds of LBP descriptors as described in Section 3.2.2 using the default neighborhood size (8, 1) and L_2 normalization. We adopted the SVM classifier. The performance results are shown in Table 3.1. We can see that LBP^{riu2} descriptor outperforms uniform LBP operators. This makes sense, as the observations of the features in frames can appear at different positions in the sequence. If we only use uniform LBP operator, we will get different LBP values for the same change pattern. Instead, LBP with rotation invariance can tackle this problem.

Table 3.1: Overall classification results using LBP^{u2} and LBP^{riu2} descriptors, presented in [66] ©2017 IEEE.

Features	Acc. (%)
MFCC & LBP^{u2}	72.5
MFCC & LBP^{riu2}	75.8

Neighborhood size (P, R)

As the size of the local patch will influence the information we get on temporal dynamics, we need to choose the neighborhood size (P, R) of the LBP^{riu2} descriptor that can encode the most useful local statistics for classification. We adopted the default L_2 normalization and the SVM classifier. We examined the results by varying the number of sampling points P and neighborhood radius R. The results are given in Table 3.2. With a neighborhood size of (2,1), the LBP encodes change patterns between adjacent frames. When we increase the neighborhood size, the LBP operates on several previous and

subsequent frames. We notice that the highest classification accuracy was found when the neighborhood radius R is 3 and the number of sampling points is 16. This reveals that the temporal information obtained from the three previous and subsequent frames is more useful for classifying scenes.

Table 3.2: Overall classification results using different neighborhood size (P, R), presented in [66] ©2017 IEEE.

P	R	Accuracy (%)
2	1	73.8
4	1	71.8
8	2	72.9
16	3	79.1

Normalization Method

The normalization method is another factor affecting the LBP features as it determines how we interpret the occurrences of different LBP patterns. To find the suitable normalization method, we examined its impact by using three types of normalization methods on the LBP^{riu2} histogram: L_1, L_2 and L_2-Hellinger normalization. The neighborhood size was set to (16, 3) and the SVM classifier was used. The L_2-Hellinger [32] normalized feature vectors can be obtained using

$$\hat{x} = \sqrt{\frac{x}{||x||_1}} \tag{3.10}$$

The performance comparison is shown in Table 3.3. The commonly used L_2 normalization outperforms both L_1 and L_2-Hellinger, achieving a recognition rate of 79.1%.

Complementary spectral features

We evaluated the performance of all the spectral features mentioned in Section 3.2.2 in charactering the spectral change over time. For each frame, we divided spectrum into three subbands, and the feature vector extracted was a concatenation of the MFCC statistics, 18-dimensional LBP^{riu2} descriptors and the averaged 3-dimensional spectral features. Table 3.4 shows the classification accuracy of the system

Table 3.3: Overall classification results using L_1, L_2, and L_2-Hellinger normalization, presented in [66] ©2017 IEEE.

Normalization method	Accuracy (%)
L_1	77.8
L_2-Hellinger	77.4
L_2	79.1

when using spectral features supplying with the MFCCs and LBP^{riu2} features. The results reveal that most of the complementary features is not useful for improving the overall system performance. This may result from the relative low dimension of the complementary spectral features compared to that of MFCCs and LBP^{riu2}. Among all these spectral features, SCF features can slightly improve the recognition accuracy, achieving a classification accuracy of 80.3%.

Table 3.4: Overall classification results when adding complementary spectral features to MFCC & LBP features, presented in [66] ©2017 IEEE.

Features	Accuracy (%)
MFCC & LBP^{riu2} & SCF	80.3
MFCC & LBP^{riu2} & SBW	79.1
MFCC & LBP^{riu2} & SF	78.7
MFCC & LBP^{riu2} & SBE	78.5
MFCC & LBP^{riu2} & SC	78.4
MFCC & LBP^{riu2} & SE	78.7
MFCC & LBP^{riu2} & RE	79.3

Comparison to baseline features using SVM and D3C

Finally, we compared the overall classification accuracy of our proposed method to that of the baseline, MFCC & RQA and MFCC & LBP in Table 3.5. The classification results using SVM and D3C are shown respectively in Fig. 3.12. The RQA features were computed using the parameters in [63]. It can

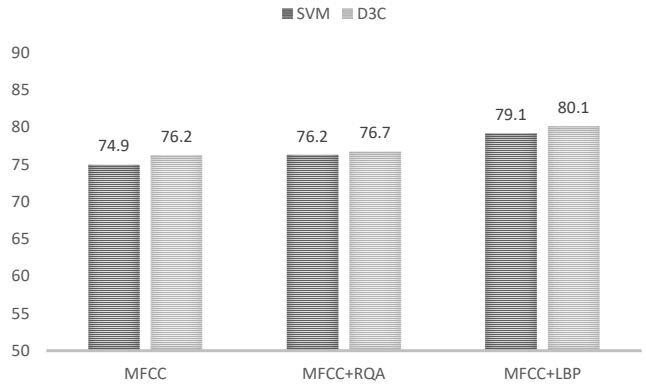


Figure 3.12: Classification accuracy using MFCC, MFCC & RQA and MFCC & LBP^{riu2} as features, SVM and D3C as classifiers, presented in [66] ©2017 IEEE.

Table 3.5: Overall classification results using MFCC, MFCC & RQA, MFCC & LBP^{riu2} and MFCC & LBP^{riu2} & SCF as features, presented in [66] ©2017 IEEE.

Features/methods	Accuracy (%)
baseline system (MFCC + GMM)	72.5
MFCC & RQA	76.7
MFCC & LBP^{riu2}	80.1
MFCC & LBP^{riu2} & SCF	80.3

be observed that the performance of ASC were improved when the temporal properties of the sound were taken into account. Our proposed features worked better than the RQA features in capturing the temporal information. In addition, the use of the ensemble classifier D3C increased the classification accuracy obtained from the SVM. The addition of spectral crest factor features further improved the ASC performance.

When we investigate the per-class accuracy by taking a look at Table 3.6, it can be observed that adding temporal features increases the recognition rate for most classes. Although the baseline system achieves better results for some classes such as Office and Metro station, the context-wise performance of our proposed system is more balanced. The most difficult scene for our system to recognize is the park, which corresponds with the baseline system.

Table 3.6: The class-wise accuracy, presented in [66] ©2017 IEEE.

Scene	Accuracy(%)	
	Baseline	Our method (%)
Beach	69.3	71.8
Bus	79.6	84.6
Cafe/restaurant	83.2	71.8
Car	87.2	88.5
City center	85.5	91.0
Forest path	81.0	97.4
Grocery store	65.0	79.5
Home	82.1	79.5
Library	50.4	87.2
Metro station	94.7	93.6
Office	98.6	93.6
Park	13.9	44.9
Residential area	77.7	66.7
Train	33.6	65.4
Tram	85.4	88.5
Total	72.5	80.3

3.4 Summary

In this Chapter, we proposed some novel features to characterize the information on temporal dynamics for ASC. This proposed method utilizes LBP descriptor to capture the temporal evolution of frame-level MFCC features. LBP features can encode this important information and can be beneficial when combining with MFCC features to improve the overall classification performance. Further, some complementary spectral features were added to the MFCC and LBP features to provide information on spectral change. In addition, we adopted the D3C classifier, which is a hybrid model that combines ensemble pruning based on k-means clustering and dynamic selection and circulating combination. The experimental results show the advantages of our proposed system. However, the proposed method is total based on the frame-based MFCC features, which did not take full advantage of the joint time-frequency characteristics. In the next chapter, this aspect is further analyzed, and a new feature extraction method is proposed for the task of SED.

Chapter 4

Sound Event Detection

In this Chapter, we focus on another challenging task of recognizing overlapping sound events simultaneously from a single channel audio signal. In addition to the problem of recognizing the general acoustic type that has been studied in the previous chapter, this problem often occurs in the multi-source environments similar to our daily life. Sound events occurring in our daily environments are rarely heard in isolation. Therefore, detecting overlapping sound events from a single continuous audio signal is quite challenging. This study introduced a new approach for SED in real-life audio using Nonnegative Matrix Factor 2-D Deconvolution (NMF2D) and RUSBoost techniques. The idea is to capture the two-dimensional joint spectral and temporal information from the time-frequency representation (TFR) while possibly separating the sound mixture into several sources. In addition, the RUSBoost technique is utilized to address the class imbalance problem of the training data.

This chapter is organized as follows. Section 4.1 provides the motivation for using NMF2D to solve the problem of SED. Section 4.2 then introduces the proposed SED system based on NMF2D features and RUSBoost ensemble techinique. Finally, experiments are conducted in Section 4.3 to evaluate the performance of the proposed system using the TUT Sound events 2016 and TUT Sound events 2017 dataset [72].

4.1 Motivation

As we mentioned in the previous chapter, temporal information can be captured across a range of different time and frequency scales, including joint spectro-temporal features that capture both spectral and temporal evolution. In the ASC domain, speech signals were found to exhibit unique time-frequency envelope patterns, and localized spectro-temporal features were demonstrated to improve the performance of ASR system [75]. There are many approaches to achieving spectro-temporal feature extraction found in the literature. An example is the use of two dimensional time-frequency filtering transformation to design a set of robust filters [76]. The authors in [77] applied both linear and nonlinear feature transformations to the logarithmic Mel-spectrum representation of speech. Transformations were based on linear discriminant analysis (LDA), independent component analysis (ICA), principal component analysis (PCA) and multilayer perceptron network based Nonlinear Discriminant Analysis (NLDA). Building upon the approach in [60], where temporal patterns were extracted from rather long (1s) and narrow (one critical band) patches, the work in [78] extended the frequency context to several critical bands and achieved higher recognition performance. In [79], a system is proposed that used two-dimensional sine-modulated Gabor filter functions to model a range of spectro-temporal patterns.

The rest of this section provides the motivation for using matrix decomposition methods to decompose the spectrogram as a set of spectro-temporal basis for sound event classication. First, problem description and an overview of the idea are given. Then, the review of previous works for SED is given. At last, the advantages and disadvantages of the most common spectrogram representations are discussed.

4.1.1 Problem Description

The problem we addressed in this Chapter is SED in real life audio. Another problem related to this is SED in synthetic audio. Although these two problems share some similarities, the datasets used to train the model are of much difference. While source-independent sound examples for each event class are provided for the task of SED in synthetic audio, the model used for SED in real life audio can only utilize relatively long audio files with overlapping sound events and less accurate event labels.

In addition, test data for SED in synthetic audio consist of synthetic mixtures of sound examples at various SNR level, event density conditions and polyphony. Contrary to this, for SED in real life audio, similar type of overlapping sound events data is used.

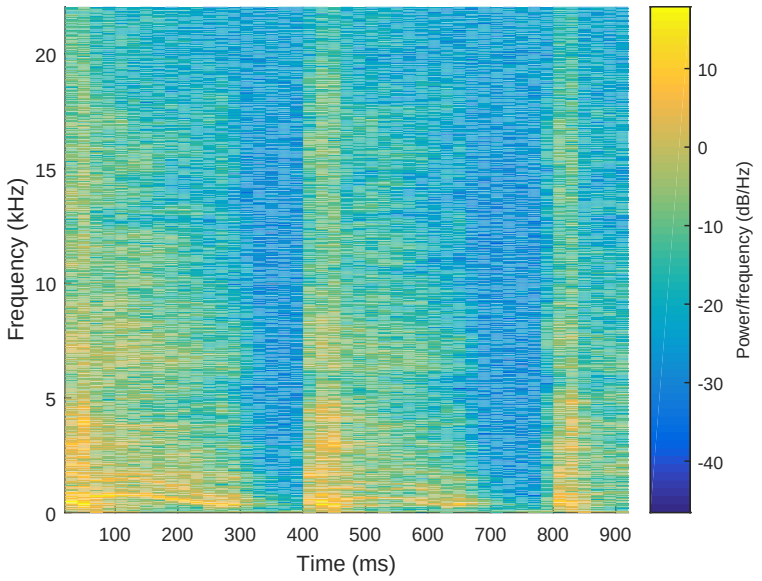
4.1.2 Overview

In addition to the "bag-of-frames" strategy mentioned in the previous chapter, the other strategy focuses on using a set of high level features to represent the scene. These features are usually captured by a dictionary of "acoustic atoms" learned from the data.

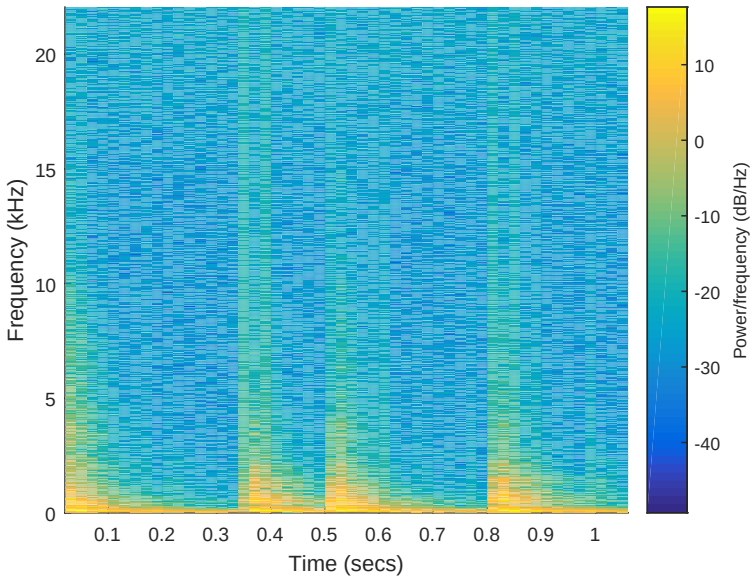
An example showing spectrograms of different sound events is given in Figure 4.1. It can be easily observed in Figure 4.1 that the spectrograms of different sound events are distinctive, thus these TFR contain information that can be utilized to detect sound events. With more careful analysis, it can be found that sound event is characterized by some repetitive time-frequency patches in the spetrograms, thus it is possible to recognize sound events using this information.

Image processing techniques are often used to derive information contained in the spectrogram. Depending on the scope of the extracted features, they can be divided in to frame-based, global and local features. Global features are derived from the whole spectrogram, thus it is difficult to combine them with the bag of frames approach. Instead, frame-based features represent characteristics of each spectrogram frame while incorporating temporal information in the feature vectors. Local features are extracted from the local time-frequency regions in the spectrogram. An example is the use of HOG to extract features about the shapes and evolution of the time-frequency structures. With temporal information being considered, the approach in [80] utilized ordered spectro-temporal patch features obtained from the Mel-spectra, and a local pooling operation across time and frequency was introduced to find the best match to the patches. Although image processing techniques are commonly used in ASC and provide state-of-the-art results, they are not suitable for SED task. As temporal positions of the local time-frequency regions are cannot be derived by image processing methods, the start and end time of each detected sound event could not be provided.

An alternative strategy related to the extraction of spectro-temporal features is to use matrix decomposition methods such as PCA, ICA and NMF. The basic idea is that the original data can be projected onto a subspace using a set of basis. The original data can then be reconstructed by a linear combination of these basis.



(a) Sound "cough"



(b) Sound knock

Figure 4.1: Example spectrograms of different sound events

4.1.3 Limitations of the current techniques

Although the topic of SED in real life audio is important and interesting, there are only a small number of previous works focusing on this problem. The approaches for SED can be divided into three categories: direct classication, matrix decomposition and computational auditory scene analysis (CASA). These approaches are introduced and discussed in next paragraphs.

Direct Classification

This method is similar to the "bag-of-frame" approach for ASC, where the popular MFCC features are used, but classification models has been modified to be suitable for the presence of overlapping sound events. An example is the use of Factorial HMMs [55]. The work in [81] utilized the conventional GMM-HMM model with a modified Viterbi decoding process. The disadvantage of this method is that the computational cost is relatively high. Another strategy is therefore to add a category containing various combinations of overlapping sound events, then perform the common classification [82]. However, this approach requires sufficient amount of the overlapped sounds in the training phase. Besides, we cannot always cover all possible combinations of overlapping sound events, thus this method is not practical enough.

Matrix Decomposition

This type of methods is inspired from blind source separation that uses factorization to decompose the input signal into its component parts. The most common approach is NMF. Without any prior information about the sound sources, the NMF separates a signal into sources in an unsupervised manner. An early example of applying NMF approach for SED was introduced in [83]. The authors employed NMF in the preprocessing step to separate the input signal into four individual tracks. MFCC features were then extracted from each of these separated signals and HMM was used to detect the sound events. The approach in [84] applied NMF to the spectrogram to decompose it into a set of templates, such that the activation of these templates during testing can be used as a indication of the present overlapping sound event classes. Additional constraints, such as sparsity, can be added to the NMF to improve the decomposition accuracy.

Computational Auditory Scene Analysis

CASA refers to the study of ASA using computational means. The concept of ASA was firstly proposed by Bregman [85] to describe the process by which the human auditory system organizes sound into perceptually meaningful elements. It is related to the cocktail party problem, where a human listener is easily able to follow a conversation with a friend in a room with many competing conversations and acoustic distractions. Different from the problem of blind signal separation, CASA is mainly based on mechanisms of the human auditory system.

The main idea of CASA is to generate a set of masks that can segment the spectrogram into regions corresponding to the different overlapping sound events. The segmentation is usually achieved by grouping the spectrogram elements based on their observed properties and cues. The main difficulty of such systems is that it is the generation of a reliable mask, since problem of mask estimation becomes more complicated for overlapping sound event sources.

4.1.4 Common Time-frequency Representations

Since our motivation is to extract distinct patterns from the time-frequency representations of the audio signal, the first step is to select a good representation. A wide range of TFR are found in the literature, and each of them may have its own advantages depending on the application. Therefore, some of these are introduced in this section.

Spectrogram

Audio signals are time-varying signals. If we take the spectrum over the whole signal, then we get the average spectrum, but will not be able to see the individual changes in fundamental frequencies. Therefore, in real-time applications we need to divide the signal into segments such that we do not have to wait for the whole audio signal to be finished before we can start processing. If we take the spectrum from small segments close to each other, we can then observe the spectral evolution of the signal. Such a representation is known as the spectrogram of a signal. When the spectrogram is calculated using windowing and the discrete Fourier transform it is called the STFT. The calculation of a spectrogram includes the following steps:

1. Apply windowing at position k to obtain segment of the signal of length N.

2. Apply the fast Fourier transform to obtain the spectrum $X_k(\omega)$ using:

$$X_k(\omega) = \sum_{n=0}^{N-1} x_k[n] e^{-i2\pi \frac{\omega}{\omega_s} n}, \tag{4.1}$$

   where $f = k\omega_s/N$ is the frequency bin for $k = 1, ..., N/2 + 1$.

3. Compress the dynamic range of the linear power spectrogram using the log function to give the conventional log power spectrogram. This is calculated as follows:

$$X_{log}(\omega) = 20 \log_{10} |X_k(\omega)|. \tag{4.2}$$

4. Advance position by K, that is, $k = k + K$ and return to the first step.

It is simple and fast to compute. However, the overall shape of the logarithmic power spectra is often a smooth shape. This means that components of the log-power-spectrum are correlated. It is not a very efficient representation of the sound. In addition, it is based on the assumption that the signal is stationary within each window, which may not be true for many sound events that have sharp discontinuities. Besides, there is a trade-off between frequency and time resolution, as choosing a longer window gives better frequency resolution at the cost of reducing the temporal resolution.

Gammatone spectrogram

This type of TFR is calculated using the cochlear filtering in the inner ear. The Gammatone filter is physiologically designed to mimic the structure of peripheral auditory processing stage, and is widely used in computational auditory models. The Gammatone impulse response describing the filter is the product of a gamma distribution function and a sinusoidal tone, and is given by

$$g(t) = at^{n-1} e^{-2\pi bt} \cos(2\pi ft + \phi), \tag{4.3}$$

where t is the time, n represents the order of the filter, ϕ is the phase of the carrier, f is the center frequency and b is the bandwidth of the Gammatone lter.

Patterson *et al.* [86] found that the impulse response of the Gammatone function of order 4 is quite suitable to the human auditory filter shapes. Glasberg and Moore [87] concluded the equivalent

rectangular bandwidth (ERB) from human data with the function:

$$ERB = 24.7(4.37 \cdot 10^{-3} f + 1). \tag{4.4}$$

When the order of the filter is 4, the value of b can be calculated using:

$$b = 1.019ERB. \tag{4.5}$$

As illustrated in Figure 4.2, a bank of Gammatone filters is commonly used to simulate the motion of the basilar membrane within the cochlea as a function of time, in which the output of each filter models the frequency response of the basilar membrane at a single place. The filterbank is normally defined in such a way that the filter centre frequencies are distributed across frequency in proportion to their bandwidth, known as the ERB scale (Glasberg and Moore, 1990). The ERB scale is approximately logarithmic, on which the filter centre frequencies are equally spaced.

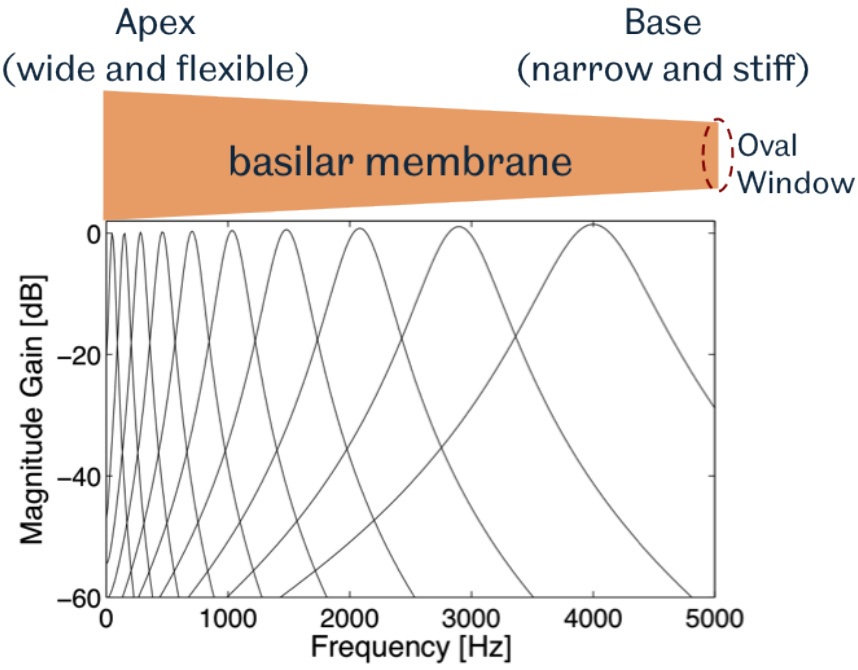


Figure 4.2: Frequency responses of a Gammatone filterbank with ten filters whose centre frequencies are equally spaced between 50 Hz and 4 kHz on the ERB-rate scale, presented in [88].

The Gammatone spectrogram has the advantage over the STFT that there is no trade-off between

time and frequency resolution. In addition, it has been shown that Gammatone filters are highly correlated with natural sound signals, which should produce a sparse, high resolution spectrogram of the sound event. The disadvantage is the relatively high computational cost, and that the common ERB scale has less resolution at higher frequencies, where a wider frequency response is used to match that found in the basilar membrane.

Mel Spectrogram

The Mel spectrogram is calculated using the Mel frequency scaling. Similar to the Gammatone lterbank, the Mel filterbank provides an approximation to the basic auditory principles of the human auditory system. It uses different resolution across frequencies and logarithmic perception of intensity. The MFCC features utilized in the previous section are calculated based on the Mel spectrogram.

Different than the Gammatone spectrogram, Mel spectrogram puts more emphasis on the efficient computation while less on biological plausibility. There, it utilizes a set of triangular-shaped filters to filter the signal in the spectral domain. As shown in Fig. 3.3, the width of each filter extends to the center frequency of the neighboring filters, hence the filters become wider at higher frequencies. Log is also commonly taken to give an equivalent spectrogram representation, Smel(f,t), where f represents the center frequencies of the Mel filters and t is the time frame of the STFT.

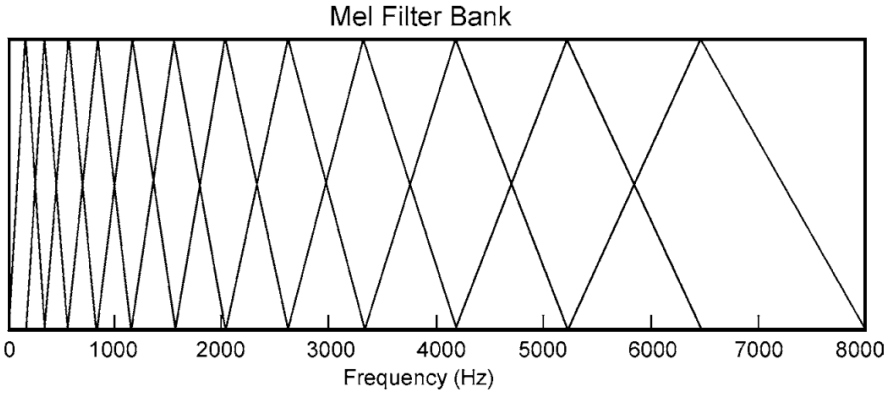


Figure 4.3: Frequency responses of a Mel filterbank with twelve filters whose centre frequencies are equally spaced between 0 Hz and 8 kHz on the Mel scale.

Constant-Q Transform

The constant-Q transform (CQT) was introduced by Brown in 1991 [89] and was used for musical representations. Similar to the FT, the CQT transforms a data series to the frequency domain. Different than the DFT that has constant difference between frequency components, CQT uses a series of logarithmically spaced filters, where the spectral width of the k-th filter is some multiple of the previous filter's width:

$$\begin{aligned} f_k =& 2^{1/n} \cdot f_0 \\ =& (2^{1/n})^k \cdot f_0, \end{aligned} \tag{4.6}$$

where f_k is the bandwidth of the k-th filter, f_0 represents the central frequency of the lowest filter, and n dictates the number of filters per octave.

Therefore, the frequency difference between the $(k+1)$-th and k-th filter is:

$$\begin{aligned} \delta f_k =& f_{k+1} - f_k \\ =& (2^{1/n} - 1) \cdot f_k. \end{aligned} \tag{4.7}$$

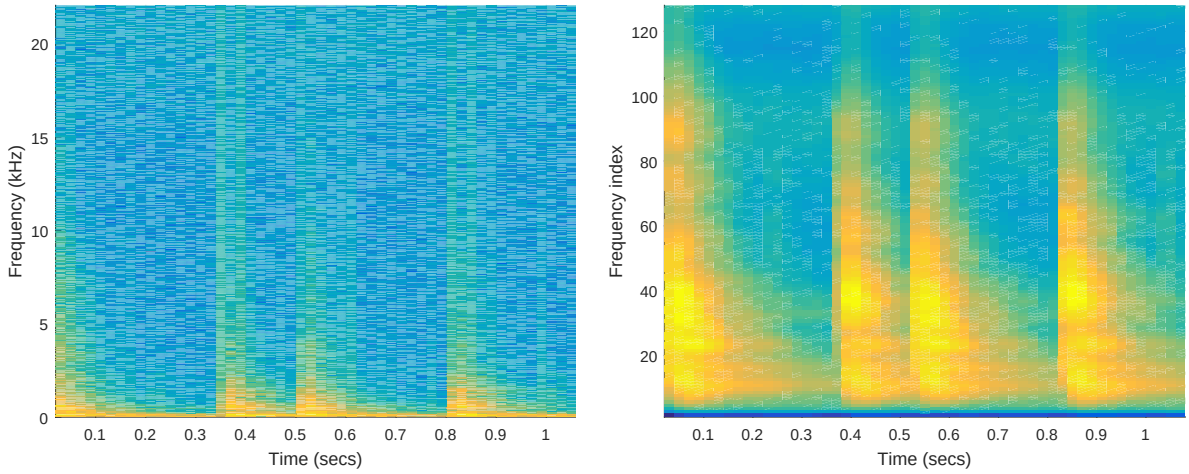
The ratio of center frequency to resolution remains a constant Q:

$$\begin{aligned} Q =& \frac{f_k}{\delta f_k} \\ =& (2^{1/n} - 1)^{-1}. \end{aligned} \tag{4.8}$$

These characteristics make the CQT useful for note identification as the center frequencies $f_k (k = 0, ...)$ can directly correspond to musical notes by using appropriate f_0 and n. In addition, CQT has increasing time resolution towards higher frequencies, which resembles the human auditory system.

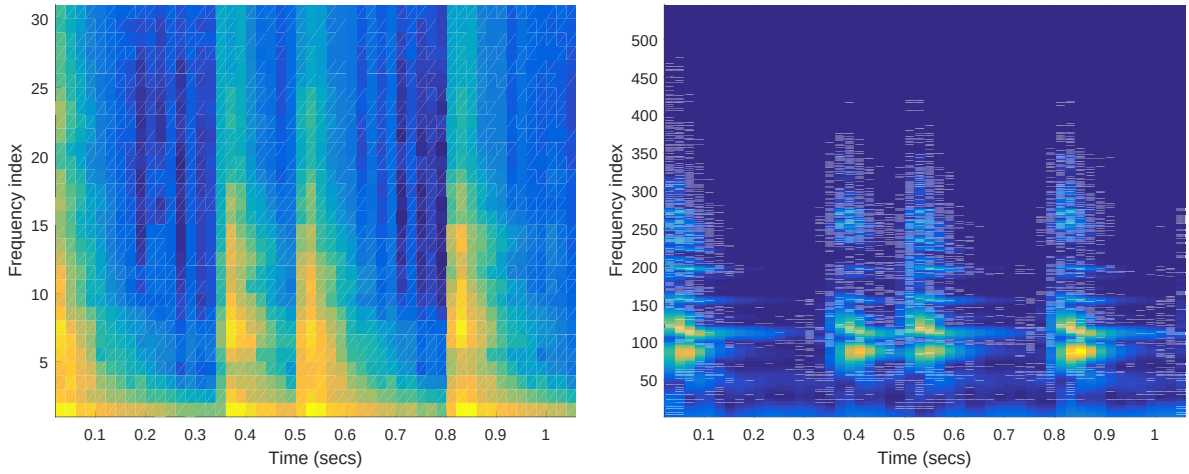
A comparison of the spectrogram, Gammatone spectrogram, Mel spectrogram and CQT representations is shown in Figure 4.4. It can be seen that each representation shows the signature of the door knocking sound while having varying characteristics. Although the spectrogram in Figure 4.4a provides the highest frequency resolution, most of the distinct information is contained in the lower frequency. However, frequency axes in the Gammatone spectrogram, Mel spectrogram and CQT representations emphasize on the lower frequencies and compress the information contained in higher frequency. How-

ever, the representation of the bells harmonic is quite different in each of the representations. In terms of harmonic structure, CQT produces different results compared to other representations.



(a) Spectrogram

(b) Gammatone spectrogram



(c) Mel spectrogram

(d) CQT

Figure 4.4: Example showing door knocking sound with different TFR

4.2 Joint spectro-temporal features for Sound Event detection

Inspired by previous works on extraction of spectro-temporal features, a novel feature extraction method for SED is now proposed. The motivation behind this idea is that TFR contains joint time-frequency patterns that characterize sound events. The proposed approach is based on the NMF2D technique that decomposes the sounds TFR into parts that contain time and frequency evolution signature. The rest of this section gives an overview of the approach, then describing the details of the joint spectro-temporal feature extraction and the classification approach.

4.2.1 Overview

Typical methods for overlapping sound event detection (SED) do not fully consider the joint spectral and temporal transition characteristics of the audio signal. They are generally based on training models using either separate data from each event class or mixed signals containing simultaneous sound events. In this study, a new approach for SED based on Nonnegative Matrix Factor 2-D Deconvolution (NMF2D) of the time-frequency representation (TFR) and RUSBoosting is proposed. The proposed approach is inspired by convolutive NMF method that has been successfully used for extracting spectro-temporal features for SED [90]. A natural extension to spectro-temporal feature extraction is to consider features that jointly model the spectral and temporal information. In NMF2D, the time-frequency (TF) signature of each sound source is modeled by the two-dimensional convolution of the spectral basis and the temporal code. This method was originally proposed for blind separation of instruments in polyphonic music [91]. A major advantage of the proposed approach is that it learns the TF characteristics of each sound source, so there is no need of estimating the number of templates required to decompose the signal. Another advantage of this method is the possibility of separating the mixture of sound events into several sources.

In addition, it is also important to localize the sound event information in time regions, thus the start and end point of each sound event can be detected. Therefore, the NMF2D approach could be applied to the TFR to characterize the distribution of joint spectro-temporal patterns in the time region.

As shown is Figure 4.5, the feature extraction process consists of the following steps:

1. Obtain the TFR of the input audio signal.

2. Time-frequency templates are learned from the TFR using the NMF2D technique.

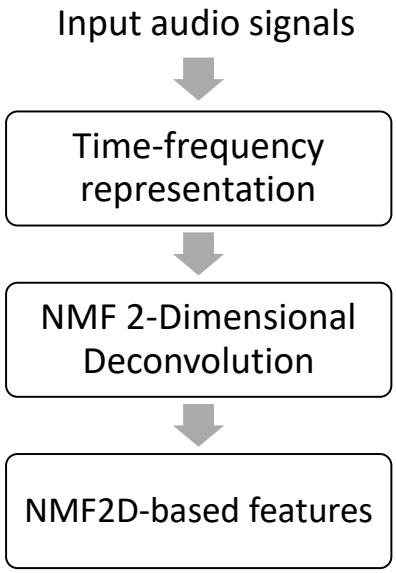


Figure 4.5: Block diagram of the proposed feature extraction

3. The activation of these templates in the time region are used as features.

Although the NMF2D technique captures a set of time-frequency templates, we still need find a reasonable way that represents the continuous activation patterns of these templates as discrete event-like features. Therefore, a sliding window of 200 ms with step size of 100 ms is used to summarize the activation patterns of time-frequency patterns by taking the log of the maximum of each activation dimension. Also, the MFCC features within the sliding window are averaged to to be combined with the NMF-based features.

For recognition, a binary classifier based on RUSBoot ensemble method is trained for detecting each sound event class using the extracted features. In the test phase, all of these binary classifiers are used simultaneously to detect their corresponding sound events, and events that are too short will be removed to produce a more accurate result.

4.2.2 Feature Extraction

Standard NMF

The standard NMF generally refers to the technique to find a low-rank approximation of a given non-negative matrix $\mathbf{V}$. Assume that $\mathbf{V} \in \mathbb{R}_+^{B \times N}$ is the matrix with size $B \times N$, the NMF problem aims to find two non-negative matrix whose product approximates the non-negative matrix $\mathbf{V}$. Let matrix $\mathbf{V}$ be the product of the matrices $\mathbf{W} \in \mathbb{R}_+^{B \times K}$ and $\mathbf{H} \in \mathbb{R}_+^{K \times N}$:

$$\mathbf{V} \approx \mathbf{WH}, \tag{4.9}$$

where the columns of $\mathbf{W}$ are basis vectors and the columns of $\mathbf{H}$ are the corresponding weights or activations. The success of the reconstruction can be measured using a variety of cost functions. Lee and Seung [92] studied two divergence functions for NMF: the squared error and Kullback-Leibler (KL) divergence. Each of them leads to a different NMF algorithm. By minimizing the cost function using gradient descent and choosing an appropriate step size, $\mathbf{W}$ and $\mathbf{H}$ can be estimated using iterative update rules. For the squared error version of NMF, $\mathbf{W}$ and $\mathbf{H}$ can be updated using:

$$\mathbf{W} \leftarrow \mathbf{W} \bullet \frac{\mathbf{VH}^T}{\mathbf{WHH}^T}, \tag{4.10}$$

$$\mathbf{H} \leftarrow \mathbf{H} \bullet \frac{\mathbf{W}^T\mathbf{V}}{\mathbf{W}^T\mathbf{WH}}. \tag{4.11}$$

For the KL divergence version of NMF, $\mathbf{W}$ and $\mathbf{H}$ can be updated using:

$$\mathbf{W} \leftarrow \mathbf{W} \bullet \frac{\frac{\mathbf{V}}{\mathbf{WH}}\mathbf{H}^T}{1 \cdot \mathbf{H}}, \tag{4.12}$$

$$\mathbf{H} \leftarrow \mathbf{H} \bullet \frac{\mathbf{W}^T\frac{\mathbf{V}}{\mathbf{WH}}}{\mathbf{W} \cdot 1}, \tag{4.13}$$

where $A \bullet B$ and $\frac{A}{B}$ denotes element-wise multiplication and division respectively.

NMF Deconvolution

The standard NMF described above are utilized for many audio tasks. However, this method cannot model the relative positions of each spectrum thus losing the temporal information. As an extension to the standard NMF, convolutive NMF was introduced by [93] to model the temporal structure of the components.

In the standard NMF the model $\mathbf{V} \approx \mathbf{WH}$ is used. In convolutive NMF the model is extended to:

$$\mathbf{V} \approx \sum_{t=0}^{T-1} \mathbf{W}_t \overset{\rightarrow t}{\mathbf{H}}. \tag{4.14}$$

The $\overset{i\rightarrow}{(\cdot)}$ operator shifts the columns of its argument by i spots to the right. For example:

$$A = \begin{bmatrix} 3 & 4 & 5 \\ 0 & 1 & 2 \\ 9 & 8 & 7 \end{bmatrix} \tag{4.15}$$

$$\overset{\rightarrow 1}{A} = \begin{bmatrix} 0 & 3 & 4 \\ 0 & 0 & 1 \\ 0 & 9 & 8 \end{bmatrix} \tag{4.16}$$

$$\overset{\rightarrow 2}{A} = \begin{bmatrix} 0 & 0 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 9 \end{bmatrix} \tag{4.17}$$

NMF 2-D Deconvolution

The NMF deconvolution model can further be extended to a 2-dimensional convolution of $\mathbf{W_t}$ which depends on time, t, and $\mathbf{H_f}$ which depends on frequency, f. The NMF2D model can be described as:

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{t=0}^{T-1} \sum_{f=0}^{F-1} \overset{\downarrow f}{\mathbf{W}}_t \overset{\rightarrow t}{\mathbf{H}}_f. \tag{4.18}$$

The $\overset{i\downarrow}{(\cdot)}$ operator moves the rows of its argument by i spots down. For example:

$$A = \begin{bmatrix} 3 & 4 & 5 \\ 6 & 1 & 2 \\ 9 & 8 & 7 \end{bmatrix} \tag{4.19}$$

$$\overset{\downarrow 1}{A} = \begin{bmatrix} 0 & 0 & 0 \\ 3 & 4 & 5 \\ 6 & 1 & 2 \end{bmatrix} \tag{4.20}$$

It can be noted that the NMFD model introduced can be considered as a special case of the NMF2D model where $F = 0$.

For the least squared error NMF2D, the cost function is given by

$$C_{LS} = \frac{1}{2}||\mathbf{V} - \mathbf{\Lambda}||_F^2. \tag{4.21}$$

$\mathbf{W}$ and $\mathbf{H}$ can be updated using:

$$\mathbf{W_t} \leftarrow \mathbf{W_t} \bullet \frac{\sum_f \mathbf{V} \overset{\uparrow f \rightarrow t}{\mathbf{H}_f}^T}{\sum_f \mathbf{\Lambda} \overset{\uparrow f \rightarrow t}{\mathbf{H}_f}^T}, \tag{4.22}$$

$$\mathbf{H_f} \leftarrow \mathbf{H_f} \bullet \frac{\sum_t \overset{\downarrow f}{\mathbf{W}_\mathbf{t}}^T \overset{\leftarrow t}{\mathbf{V}}}{\sum_f \overset{\downarrow f}{\mathbf{W}_\mathbf{t}}^T \overset{\leftarrow t}{\mathbf{\Lambda}}}. \tag{4.23}$$

For the NMF2D by KL Divergence, the cost function is given by

$$C_{LS} = \sum_{i,j} \mathbf{V}_{i,j} \log \frac{\mathbf{V_{i,j}}}{\mathbf{\Lambda_{i,j}}} - \mathbf{V_{i,j}} + \mathbf{\Lambda_{i,j}}. \tag{4.24}$$

$\mathbf{W}$ and $\mathbf{H}$ can be updated using:

$$\mathbf{W_t} \leftarrow \mathbf{W_t} \bullet \frac{\sum_f (\frac{\overset{\uparrow f}{\mathbf{V}}}{\mathbf{\Lambda}}) \overset{\rightarrow t}{\mathbf{H}_f}^T}{\sum_f 1 \cdot \overset{\rightarrow t}{\mathbf{H}_f}^T}, \tag{4.25}$$

$$\mathbf{H_f} \leftarrow \mathbf{H_f} \bullet \frac{\sum_t \overset{\downarrow f}{\mathbf{W_t}}^T \overset{\leftarrow t}{(\frac{\mathbf{V}}{\mathbf{\Lambda}})}}{\sum_f \overset{\downarrow f}{\mathbf{W_t}}^T \cdot 1}. \tag{4.26}$$

An example showing the CQT of three isolated sound events and their mixture is given in Figure 4.6. NMF2D algorithm is then applied on the CQT of their mixture to find the time-frequency signature characterizing each sound events. For the NMF2D we used three, since we seek to separate three sound sources. We chose to use 5 convolutive components in time and 20 convolutive components in frequency.

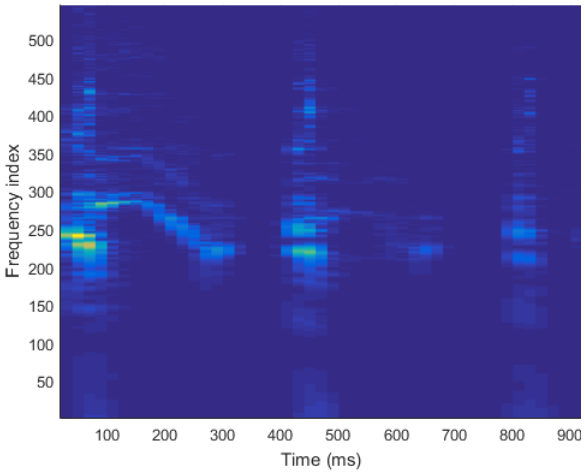
The NMF2D representation of three overlapping sound events is given in Figure 4.7 (Audio files available at https://drive.google.com/file/d/0B5tB5LTOhACqQ3JCZWZaSU00dHM/view?usp=sharing). It can be noted that three templates characterizing the time-frequency signature of each sound event are successfully learned from the CQT, which means overlapping sound can be separated into three components. On the other hand, spectral change information of these templates is also included in the activation patterns. It can be seen that the activation matrix of each template is to some extend similar to the CQT of a isolated sound, therefore these activation patterns should be useful to identify the corresponding sound events.

4.2.3 Detection with RUSBoost Ensemble technique

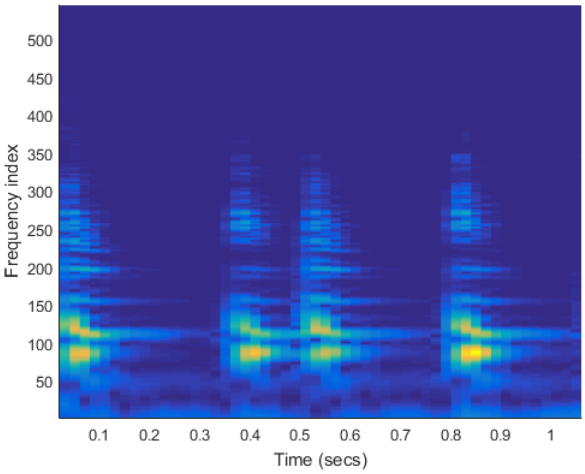
A major problem of the detection-by-classification scheme for SED is the class imbalance among different sound events, since the available instances for each event vary. For the methods where a binary classifier is used to identify each sound event, the background class usually have much more examples than the event class. Therefore, the conventional classifiers cannot produce satisfying results. Techniques to alleviate the problem of imbalanced training data are introduced in this section.

Data Sampling

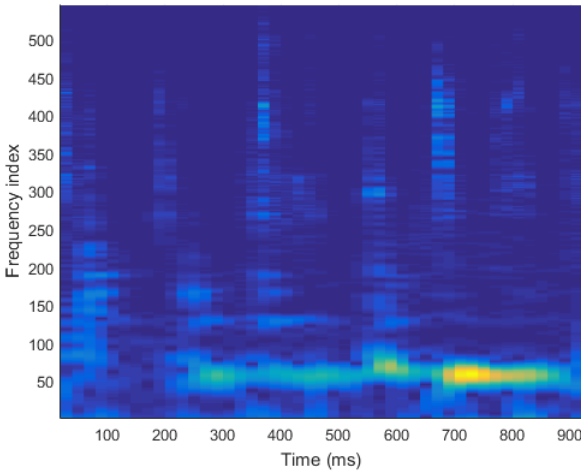
Data sampling is a basic technique to adjust the class distribution in the training data. It can be divided into two categories: oversampling and undersampling. While oversampling is to add more examples to the minority class, undersampling instead aims at removing some examples from the majority class. Random sampling is one of the simplest methods for resampling a dataset. It can be used for both oversampling and undersampling by randomly duplicating or removing examples until a desired class



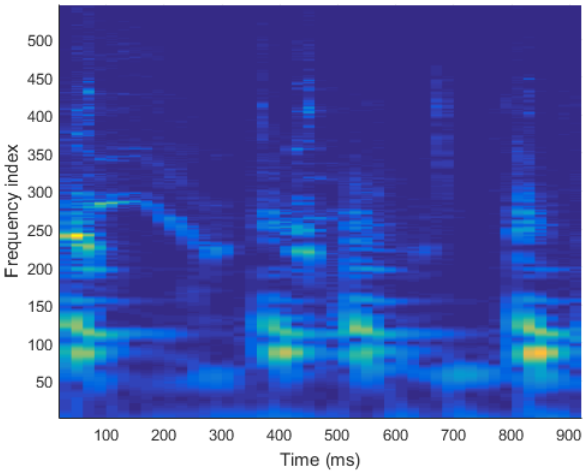
(a) Isolated coughing sound



(b) Isolated knocking sound



(c) Isolated keyboard sound



(d) Mixture of coughing, knocking and keyboard sounds

Figure 4.6: CQT of isolated sounds and their mixture

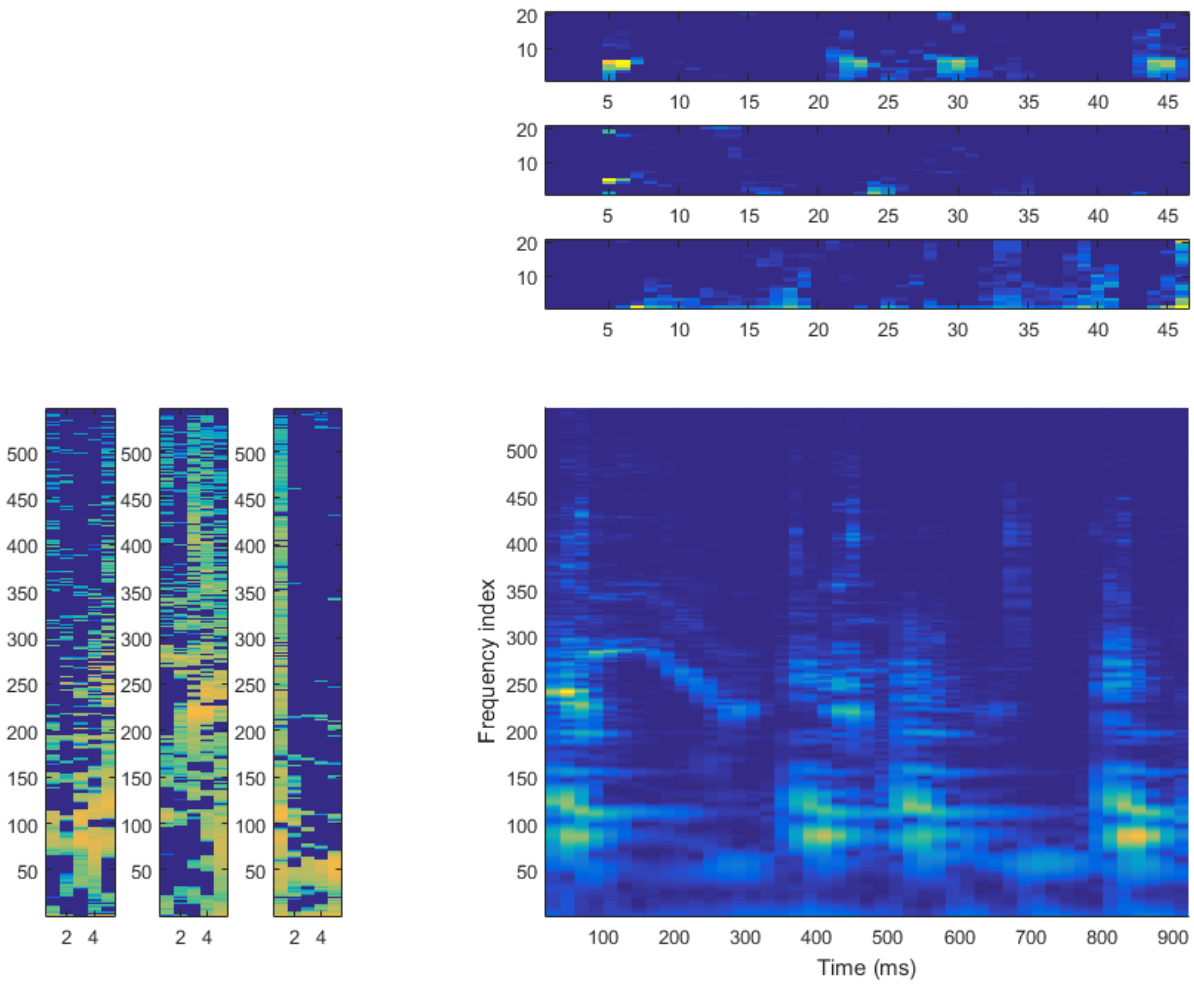


Figure 4.7: Factorization of the mixture of sound events using NMF2D. The three time-frequency plots on the left are W_t that represents time-frequency signature for each factor. The two time-frequency change plots on the top are H_f for each factor showing how these joint spectro-temporal features are placed in time and frequency.

ratio is achieved. Although undersampling reduces the training complexity, some information in the training data will be lost. On the other hand, oversampling keeps all the information with increased training time, but it can lead to overfitting.

Boosting

Boosting is another type of technique that can be used to improve classification accuracy of skewed data. Different from the data sampling techniques that are designed to address the imbalanced data in the training set, boosting aims at constructing a strong classifier as linear combination of weak classifiers. An example is the AdaBoost [94] classifier, which adjusts sample weights optimally by focusing on data points that have been misclassified by the previous weak classifier. It then combines these weak classifiers into a unified prediction by using an optimally weight majority vote of their outputs. During each iteration, those misclassified observations have their weights increased. This process continues to add weak learner until a predefined threshold is reached. Given an imbalanced dataset, samples in the minority class are most likely to be classified incorrectly and they will receive more weights in subsequent iterations. Therefore, it might improve the classification accuracy of minority class.

RUSBoost Ensemble technique

RUSBoost [17] is a hybrid algorithm that combines random undersampling and boosting techniques. In this method, boosting is performed by resampling other than reweighting to adapt algorithms that cannot incorporate example weights in their training processes. In each iteration, the majority class is undersampled to train a new model, and finally to improve the classification performance of the minority class. Combining RUS with the boosting process can decreases time needed to train a model, thus it is beneficial for learning from skewed dataset.

4.3 Experiments

In this section, different experiments are conducted to compare the performance of the proposed method. The RUSBoost ensemble classifier with decision trees are used for detection/classification.

4.3.1 Dataset

The dataset used for the evaluation is the TUT Sound events 2016 development set [72] and TUT Sound events 2017 development set [72]. TUT Sound events 2016 is a subset of TUT Acoustic scenes 2016 dataset used in the previous section. This dataset consists of audio recording from two acoustic scene: home and residential area. Each recording was captured in a different location: different streets and different homes. For each recording location, a 3-5 minute long audio recording was captured. There are 12 recordings contained in home sound events data, and 10 recordings contained in residential area sound events data. For each acoustic scene, the selected sound events is shown in Figure 4.8.

The TUT Sound Events 2017 dataset consists of recordings of street acoustic scenes with various levels of traffic and other activity. The scene was selected as representing an environment of interest for detection of sound events related to human activities and hazard situations. Each recording was captured in a different location. For each recording location, a 3-5 minute long audio recording was captured. There are 24 recordings contained in data.The selected sound events is shown in Figure 4.9.

In home scenes, the selected events are mostly abstract object impact sounds such as dishes, cutlery, and so on. While the selected sound event classes in residential area scenes are mostly related to individual physical sound sources such as bird singing, children shouting, wind blowing.

4.3.2 Evaluation metrics

There are generally two types of metrics: segment-based and event-based metrics. Segment-based evaluation is done in a fixed time block, using segments of one second length to compare the ground truth and the system output. However, event-based evaluation is done in each event. An event in the system output is considered correctly detected if its temporal position is of certain degree of overlapping with the temporal position of an event with the same label in the ground truth. A tolerance is allowed for the onset and offset. The exact description of these metrics presented in [95]. The metrics used in this study are introduced in the next paragraphs.

For segment based evaluation, the following measures are counted in each segment k:

- true positives TP: events indicated as active by both the ground truth and system output.
- false positives FP: events indicated as active by the system output but inactive by the ground truth.

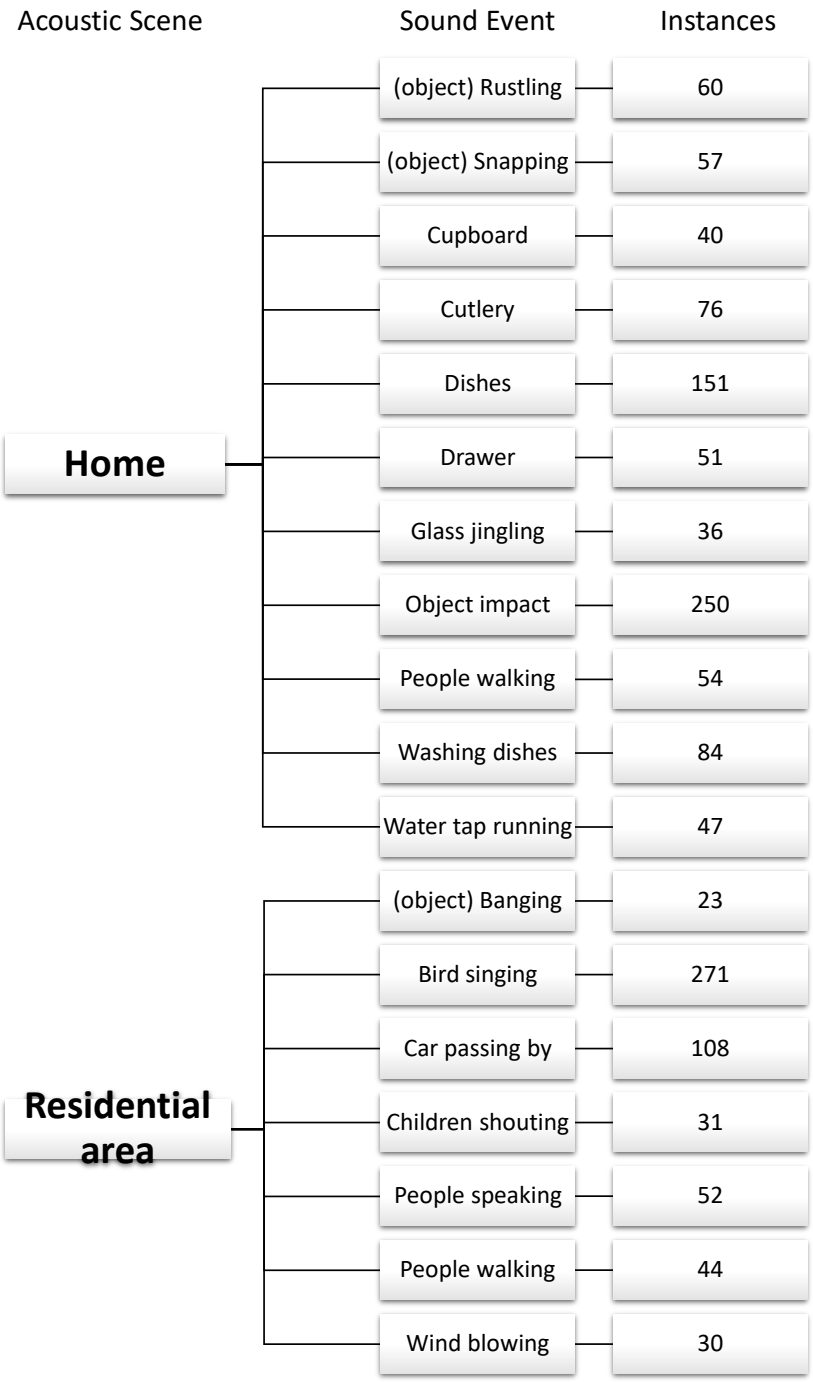


Figure 4.8: TUT SOUND EVENTS 2016 dataset

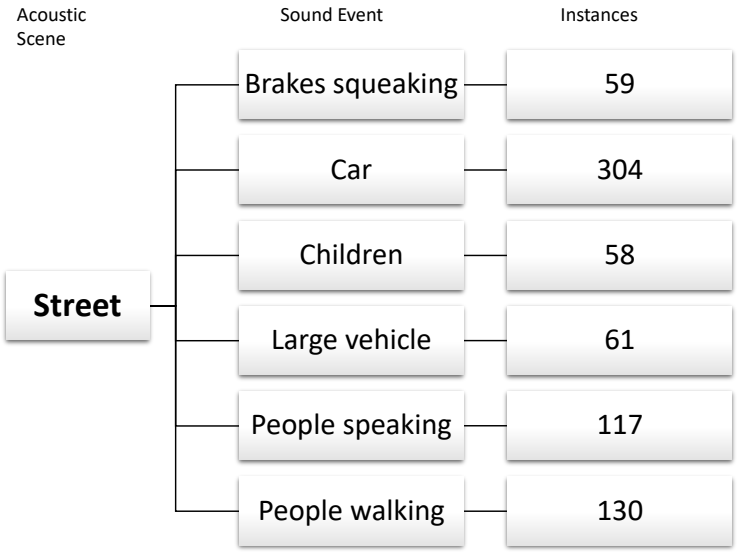


Figure 4.9: TUT SOUND EVENTS 2017 dataset

- false negatives FN: events indicated as inactive by the system output but active by the ground truth.
- substitutions S: system output indicating as active a wrong label events; one substitution is equivalent to one false positives and one false negative, meaning the system did not detect the correct event but detected something.
- insertions I: false positives after subtracting the substitutions.
- deletions D: false negatives after subtracting the substitutions.
- reference events N: number of events in the ground truth.

Error Rate

In the DCASE challenge, segment based error rate is used for the evaluation of sound event detection. This metric focuses on evaluating the events detected in non-overlapping segments, and is mainly used for applications that require fairly coarse time resolution, where more importance is placed into detecting the right events within the segment than finding their temporal position. Error rate is calculated over

all test data based on the total number of insertions, deletions and substitutions using:

$$ER = \frac{\sum S(k) + \sum D(k) + \sum I(k)}{\sum N(k)}, \tag{4.27}$$

where S is the number of substitutions representing system output indicating as active a wrong label events, D is the number of deletions, i.e false positives after subtracting the substitutions, I is the number of deletions, i.e false positives after subtracting the substitutions, and N is the total number of events in each segment k.

F-score

Accuracy in each segment is given by the F-score, based on precision and recall, which are calculated in each segment. Precision is defined as the number of correctly detected sound events divided by the total number of events detected. Recall is calculated using the number of correctly detected sound events divided by the total number of events in the ground truth. F-score is calculated over all test data based using:

$$F = \frac{2P \cdot R}{P + R} \tag{4.28}$$

where

$$P = \frac{\sum TP(k)}{\sum TP(k) + \sum FP(k)}, \quad R = \frac{\sum TP(k)}{\sum TP(k) + \sum FN(k)}, \tag{4.29}$$

The advantage of using F-score to evaluate SED performance is that it is widely known and easy to understand. However, the magnitude of F-score is mainly determined by the number of true positives such that majority classes might dominate minority classes in instance-based method [96].

4.3.3 Experimental Setup

Cross validation setup

Both of the TUT Sound events 2016 and TUT Sound events 2017 set are partitioned into four folds, such that each recording is used exactly once as test data. The only constraint imposed in this stage was that the test set cannot contain classes unavailable in training set.

NMF2D Evaluation Methods

Since the NMF2D algorithm can be applied to different TFR, and the number of convolutive components in time and frequency will affect the structure of the learned templates, the following experiments are conducted to investigate these factors in generating the NMF2D-based features:

1. Gammatone spectrogram vs. Mel spectrogram vs. CQT
2. The number of temporal convolutive components
3. The number of spectral convolutive components

Baseline Method

The baseline system for TUT Sound Events 2016 dataset is based on MFCC+GMM approach. The acoustic features include MFCC static coefficients, delta coefficients and delta-delta coefficients. For each event class, a binary classifier is set up. The class model is trained using the audio segments annotated as belonging to the modeled event class, and a negative model is trained using the rest of the audio. The decision is based on likelihood ratio between the positive and negative models for each individual class, with a sliding window of one second.

The baseline system for TUT Sound Events 2017 is based on a multilayer perceptron architecture with log mel-band energies as features. Mel-band energies are further processed using a 5-frame context, resulting in a feature vector length of 120. Using these features, a neural network containing two dense layers of 50 hidden units per layer and 20% dropout is trained for 200 epochs for each class. Detection decision is based on the network output layer containing sigmoid units that can be active at the same time. A detailed description is available in the baseline system documentation.

4.3.4 Results and Discussion

The results obtained from the experiments on the proposed SED system are now presented and compared with the baseline methods. First, some key factors that contribute to the success of the NMF2D-based method are explored: the type of TFR and the number of convolutive components in time and frequency. Then, the performance of the best performing NMF2D-based method is compared to results achieved by the baseline method.

Table 4.1: Detection results of the proposed method, exploring the different TFR that contribute to give the best performance. The computational costs of these TFRs are also listed. Except for the number of frequency bins b, the value of other factors is fixed in this evaluation. T and F represents the the number of convolutive components in time and frequency respectively, K is the number of templates and N is the number of frames.

TFR type	Computational cost	Result	
	O(K(bT+FN))	ER	F-score
Gammatone spectrogram	b = 64	0.91	41.7%
Mel spectrogram	b = 40	0.86	35.7%
CQT	b = 545	0.85	35.3%

Gammatone Spectrogram vs. Mel Spectrogram vs. CQT

The results obtained from Gammatone spectrogram, Mel spectrogram and CQT are used to analyze the effect of the TFR. For NMF2D, the number of convolutive components in time and frequency are both set to 5, and the number of factors is set according to the number of sound events contained in each acoustic scene. In this case, the computational cost of NMF2D is almost linear to the number of frequency bins. It can be seen in Table 4.1 that the CQT outperforms the other two representations in both metrics. This indicates that NMF2D is more useful to decompose representation for which frequency shift make sense. On the other hand, Mel spectrogram also produced competitive results with a reduced computational cost. Therefore, we choose to use the Mel spectrogram as the TFR of the audio signal.

The number of spectral and temporal convolutive components

An comparison to examine is how the effect of the number of convolutive components. Table 4.2 shows that the NMF2D performs well when 5 spectral and temporal components are used. This can be explained by the way in which the local time-frequency pattern shows in the spectrogram, commonly with a time region of about 100ms (5 frames) and a frequency shift of several bins.

Table 4.2: Detection results of the proposed method, exploring the number of convolutive components in time and frequency that contribute to give the best performance. TUT 2016 Sound events dataset contains 12 recordings for home sound events, and 10 recordings for residential area sound events. Each recording is 3-5 minute long.

# of temporal convolutive components	# of spectral convolutive components	Result	
		ER	F-score
5	5	0.86	37.5%
5	10	0.91	41.0%
10	5	0.88	40.6%
10	10	0.94	22.8%

Comparison to Baseline Method

The performance of the proposed approach is first compared to the baseline method provided with the TUT Sound events 2016 dataset. From the results in Table 4.3, it can be seen the proposed method outperforms the baseline MFCC+GMM method, with reduced error rate and higher F-score. Results obtained from the TUT Sound events 2017 dataset also show that the proposed approach can reduce the error rate by 3% while increasing the F-score by 8.1%.

Table 4.3: Detection results comparing with the baseline method using TUT Sound events 2016 dataset, which contains 12 recordings for home sound events, and 10 recordings for residential area sound events. Each recording is 3-5 minute long.

Method	Average		Home		Residential Area	
	ER	F-score	ER	F-Score	ER	F-score
Baseline	0.91	23.7%	0.96	15.9%	0.86	31.5%
Proposed approach	0.86	37.5%	1.06	21%	0.67	54.0%

4.4 Summary

This chapter introduced the idea of using NMF2D and RUSBoost techniques for SED in real life audio. The idea is to capture the two-dimensional spectro-temporal information in the TFR while separating the sound source into several components. Existing methods for extracting joint spectro-temporal information were first reviewed. Motivated by these approaches, the NMF2D method was then proposed for overlapping sound, which decompose the TFR of the signal into time-frequency templates to produce a set of features containing the spectral and temporal evolution information. Finally, RUSBoost ensemble technique was adopted in the detection phase to overcome the class imbalance in the training data. The results demonstrated that the proposed system comprising Mel spectrogram, NMF2D and RUSBoost approaches performed better than the baseline methods in different acoustic scenes.

Chapter 5

Conclusions and Future work

This thesis addressed the problem of ESR, which generally includes two task: classification of acoustic scenes and detection of sound events. This is a challenging task as the environmental sounds often contains components from unknown sources, and the noise level varies in different recordings. Based on the two main categories of features for ESR, this thesis has developed novel approaches to capture the temporal information from the signal. The motivation behind these approaches is that temporal information can be extracted over across a range of different time and frequency scales. To give a conclusion of this study, Section 5.1 summarizes the contributions of this thesis. Then, discussions are given in Section 5.2 regarding some of the future directions that can be explored, and the challenges that are faced.

5.1 Contributions

The approaches proposed this thesis focused on capturing temporal dynamics from different time and frequency extent. By taking this two-dimensional approach, the extracted features naturally capture both spectral and temporal information together. This is different from conventional frame-based audio features, which typically extract a feature that represents only the spectral information contained within each short-time frame. By combining image processing-inspired feature extraction from the spectrogram, with techniques for noise robust recognition, the resulting methods can significantly improve upon the state-of-the-art methods across a range of challenging experimental conditions. This idea of using spec-

trogram image processing has formed This idea of using spectrogram image processing has formed the basis for the contributions presented in this thesis, which are summarised below.

5.1.1 Combining Temporal Features by Local Binary Pattern

For ASC, the goal is to recognize the general acoustic scene indicating the location of the recording using some global features extracted from the whole audio signal. Therefore, the idea of the proposed approach is to extract the temporal signature from the frame-based MFCC, with the work published in IEEE/ACM Transactions on Audio, Speech and Language Processing [**?**]. The method first extracts the sub-band MFCC features. Then, these frame-level features are viewed an image, and LBP is then extracted from it to characterize the pixel changes in the image. This can capture the temporal evolution information contained in the sound signal, hence can improve the performance of the frame-based features found in conventional audio processing systems. In the classification, D3C ensemble classifier is adopted to combine the outputs of several base classifiers.

Experiments were first carried out to compare the different aspects that contributed the best performance of LBP using 15 audio scene classes from the TUT Acoustic Scene 2016 dataset. These included varying the type of LBP and neighborhood range used, and performing different normalization methods for LBP. The results demonstrated that the LBP^{riu2} descriptor outperforms uniform LBP operators. Additionally, it was found that the highest classification accuracy was achieved when the neighborhood radius R is 3 and the number of sampling points is 16. This reveals that the temporal information obtained from the three previous and subsequent frames is more useful for classifying scenes. And commonly used L_2 normalization method is selected as it outperforms both L_1 and L_2-Hellinger. Then, some spectral features are added to supply with the MFCCs and LBP^{riu2} features in attempt to further improving the performance. The results reveal that most of the complementary features is not useful for improving the overall system performance. Among all these spectral features, SCF features can slightly improve the recognition accuracy, achieving a classification accuracy of 80.3%. Also, experiments are conducted to compare the proposed system with the baseline method as well as some different techniques addressed the temporal dynamics. The results showed that the proposed method achieved an 8% improvement of as compared with the baseline.

5.1.2 Extracting Joint Spectro-temporal Features by NMF 2-D Deconvolution

For SER, the goal is to detect the presence of individual sound events from a relatively long audio signal. Therefore, the idea of the proposed approach is to extract the joint spectro-temporal signature from the TFR. First, the TFR of the signal is obtained from short-time spectral analysis. NMF2D is then applied on the TFR to learn the time-frequency templates. The activation of these templates in the time region can be used as features. With the combination of spectro-temporal information contained in the templates and the indication of the frequency change in the activation patterns, the joint spectro-temporal characteristics can be captured while separating the overlapping signal into several components. On the other hand, RUSBoost technique is utilized to address the class imbalance problem.

Experiments were first carried out to compare the different aspects that contributed the best performance of NMF2D using TUT Sound Event 2016 dataset. These included varying the type of TFR, and the number of spectral and temporal convolutive components. The results demonstrated that the CQT outperforms the other selected TFRs. However, we chose the Mel spectrogram instead as it would lead to a relatively low computational cost, and also provided a good result. Additionally, it was found that NMF2D performs well when 5 spectral and temporal components are used. This may result from the way in which the local time-frequency pattern shows in the spectrogram, commonly with a time region of about 100ms (5 frames) and a frequency shift of several bins. Also, experiments are conducted to compare the proposed system with the baseline method using the TUT Sound Event 2016 and TUT Sound Event 2017 dataset. The results showed that the proposed method outperformed the baseline methods. For the TUT Sound Event 2016 dataset, the proposed method reduced the total error rate by 5% whilst increasing the F1 score by 13.8%. For the TUT Sound Event 2017 dataset, the proposed method reduced the total error rate by 3% whilst increasing the F1 score by 8.1%. While the proposed method can improve the baseline performance, the result is still not satisfying.

5.2 Future work

The goal of this thesis is to develop novel features for ESR and improve its performance. This has resulted in the development of two techniques that address the extraction of temporal information and the employment of ensemble learning techniques. Although experimental results show that the

performance of the proposed methods can exceed the baseline methods, it still need to be improved in order to compete with top ranked methods that are based on deep learning. For ASC, we only utilized the frame-based features, so features from time-frequency aspects can still be explored. Image processing techniques can be applied on the TFR to extract some global features of the whole signal. For the classification approach, the ensemble of deep learning models might produce better classification result.

For SED, local features from the field of image processing may be effective for the recognition of individual sound event. In the proposed method, NMF2D is performed to decompose the signal. However, the optimal solution is, in most cases, non-unique and the problem is ill-posed. Additional constraints (e.g., sparsity, smoothness, spatial information.) can be imposed to improve the performance. In addition, a verification process can be further added to verify the detected sound events, thus reduce the number of insertions. Other possible improvements of the current method for SED include: estimating the noise distribution in the TFR, learning the time-frequency templates using supervised method and trying different base classifiers to be combined using the RUSBoosting technique. We believe these will enable this promising method to be applied in applications with a larger range of sound classes and real-world conditions.

List of Acronyms

ASA Acoustic Scene Analysis.

ASC Acoustic Scene Classification.

AW MPEG-7 Audio Waveform.

CASA Computational Auditory Scene Analysis.

CNN Convolutional Neural Networks.

CQT Constant-Q Transform.

DCASE Detection and Classification of Acoustic Scenes and Events.

DNN Deep Neural Networks.

ESR Environmental Sound Recognition.

FT Fourier Transform.

GMMs Gaussian Mixture Models.

HMMs Hidden Markov Models.

HOG Histogram of Gradients.

ICA Independent Component Analysis.

ILD Interaural Level Difference.

ITD Interaural Time Difference.

LBP Local Binary Pattern.

LPCC Linear Prediction Cepstrum Coefficients.

MFCC Mel-frequency Cepstral Coefficients.

MFCs Mel-scaled Filter-bank Coefficients.

MP Matching Pursuit.

MP-TFD Matching Pursuit Time-frequency Distribution.

NMF Non-negative Matrix Factorization.

NMF2D Non-negative Matrix Factor 2-D Deconvolution.

PLCA Probabilistic Latent Component Analysis.

RDNN Recurrent Deep Neural Networks.

RE Renyi Entropy.

RNN Recurrent Neural Networks.

SBE Spectral Band Energy.

SBW Spectral Bandwidth.

SC Spectral Centroid.

SCF Spectral Crest Factor.

SE Shannon Entropy.

SED Sound Event Detection.

SF Spectral Flatness.

STE Short-time Energy.

STFT Short Time Fourier Transform.

SVM Support Vector Machine.

TDOA Time Difference of Arrival.

TFM Time-frequency Matrix.

TFR Time-frequency Representation.

WT Wavelet Transform.

ZCR Zero-crossing Rate.

Appendix 1

Reproduction Permission

Title: Combining Temporal Features by Local Binary Pattern for Acoustic Scene Classification
Author: Wenjun Yang
Publication: Audio, Speech, and Language Processing, IEEE/ACM Trans on (T-ASL)
Publisher: IEEE
Date: June 2017

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK ››› CLOSE WINDOW

References

[1] R. F. Lyon, "Machine hearing: An emerging field [exploratory DSP]," *IEEE Signal Processing Magazine*, vol. 27, pp. 131–139, Sept. 2010.

[2] R. Pieraccini and L. Rabiner, *The voice in the machine: building computers that understand speech.* MIT Press, 2012.

[3] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech communication*, vol. 52, no. 1, pp. 12–40, 2010.

[4] F. Wang, X. Wang, B. Shao, T. Li, and M. Ogihara, "Tag integrated multi-label music style classification with hypergraph.," in *ISMIR*, pp. 363–368, 2009.

[5] E. Benetos, M. Kotti, and C. Kotropoulos, "Musical instrument classification using non-negative matrix factorization algorithms and subset feature selection," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 5, pp. V–V, IEEE, 2006.

[6] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "A survey of audio-based music classification and annotation," *IEEE transactions on multimedia*, vol. 13, no. 2, pp. 303–319, 2011.

[7] D. Wang and G. J. Brown, *Computational auditory scene analysis: Principles, algorithms, and applications.* Wiley-IEEE Press, 2006.

[8] D. Gerhard, *Audio signal classification: History and current techniques.* Citeseer, 2003.

[9] N. Yamakawa, T. Kitahara, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno, "Effects of modelling within-and between-frame temporal variations in power spectra on non-verbal sound

recognition," in *11th Annual Conference of the International Speech Communication Association: Spoken Language Processing for All, INTERSPEECH 2010*, 2010.

[10] F. Alías, J. C. Socoró, and X. Sevillano, "A review of physical and perceptual feature extraction techniques for speech, music and environmental sounds," *Applied Sciences*, vol. 6, no. 5, p. 143, 2016.

[11] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, pp. 16–34, May 2015.

[12] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*, pp. 85–90, IEEE, 1994.

[13] K. Carola Wagener, M. Hansen, and C. Ludvigsen, "Recording and classification of the acoustic environment of hearing aid users," *Journal of the American Academy of Audiology*, vol. 19, no. 4, pp. 348–370, 2008.

[14] C. Landone, J. Harrop, and J. Reiss, "Enabling access to sound archives through integration, enrichment and retrieval: The easaier project.," in *ISMIR*, pp. 159–160, 2007.

[15] L. Zu, P. Yang, Y. Zhang, L. Chen, and H. Sun, "Study on navigation system of mobile robot based on auditory localization," in *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pp. 321–326, IEEE, 2009.

[16] C. Lin, W. Chen, C. Qiu, Y. Wu, S. Krishnan, and Q. Zou, "LibD3C: ensemble classifiers with a clustering and dynamic selection strategy," *Neurocomputing*, vol. 123, pp. 424–435, 2014.

[17] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Rusboost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 1, pp. 185–197, 2010.

[18] A. Rabaoui, M. Davy, S. Rossignol, and N. Ellouze, "Using one-class svms and wavelets for audio surveillance," *IEEE Transactions on information forensics and security*, vol. 3, no. 4, pp. 763–775, 2008.

[19] G. Muhammad and K. Alghathbar, "Environment recognition from audio using mpeg-7 features," in *Embedded and Multimedia Computing, 2009. EM-Com 2009. 4th International Conference on*, pp. 1–6, IEEE, 2009.

[20] X. Valero and F. Alías, "Applicability of mpeg-7 low level descriptors to environmental sound source recognition," in *Proceedings 1st Euroregio Conference, Ljubjana*, 2010.

[21] V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi, and T. Sorsa, "Computational auditory scene recognition," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, vol. 2, pp. II–1941, IEEE, 2002.

[22] D. Hosseinzadeh and S. Krishnan, "On the use of complementary spectral features for speaker recognition," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, p. 46, 2008.

[23] S. Esmaili, S. Krishnan, and K. Raahemifar, "Content based audio classification and retrieval using joint time-frequency analysis," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '04)*, vol. 5, pp. V–665–8 vol.5, May 2004.

[24] B. Clarkson, N. Sawhney, and A. Pentland, "Auditory context awareness via wearable computing," in *IN PROCEEDINGS OF THE 1998 WORKSHOP ON PERCEPTUAL USER INTERFACES (PUI98*, 1998.

[25] 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.

[26] N. Sawhney and P. Maes, "Situational awareness from environmental sounds," 1997.

[27] R. D. Patterson and J. Holdsworth, "A functional model of neural activity patterns and auditory images," *Advances in speech, hearing and language processing*, vol. 3, no. Part B, pp. 547–563, 1996.

[28] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *the Journal of the Acoustical Society of America*, vol. 55, no. 6, pp. 1304–1312, 1974.

[29] S. Chu, S. Narayanan, and C. C. J. Kuo, "Environmental sound recognition with time –frequency audio features," *and Language Processing IEEE Transactions on Audio, Speech*, vol. 17, pp. 1142–1158, Aug. 2009.

[30] W. Nogueira, G. Roma, and P. Herrera, "Sound scene identification based on mfcc, binaural features and a support vector machine classifier," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, 2013.

[31] S. Adavanne, G. Parascandolo, P. Pertilä, T. Heittola, and T. Virtanen, "Sound event detection in multichannel audio using spatial and harmonic features," 2016.

[32] T. Kobayashi and J. Ye, "Acoustic feature extraction by statistics based local binary pattern for environmental sound classification," in *Proc. Speech and Signal Processing (ICASSP) 2014 IEEE Int. Conf. Acoustics*, pp. 3052–3056, May 2014.

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

[34] R. Hennequin, R. Badeau, and B. David, "Nmf with time –frequency activations to model non-stationary audio events," *and Language Processing IEEE Transactions on Audio, Speech*, vol. 19, pp. 744–753, May 2011.

[35] B. Ghoraani and S. Krishnan, "Time –frequency matrix feature extraction and classification of environmental audio signals," *and Language Processing IEEE Transactions on Audio, Speech*, vol. 19, pp. 2197–2209, Sept. 2011.

[36] E. Benetos, M. Lagrange, and S. Dixon, "Characterisation of acoustic scenes using a temporally-constrained shift-invariant model," in *15th International Conference on Digital Audio Effects Conference (DAFx-12)*, 2012.

[37] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[38] M. Cowling and R. Sitte, "Comparison of techniques for environmental sound recognition," *Pattern recognition letters*, vol. 24, no. 15, pp. 2895–2907, 2003.

[39] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on signal processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[40] K. Umapathy, S. Krishnan, and S. Jimaa, "Multigroup classification of audio signals using time-frequency parameters," *IEEE Transactions on Multimedia*, vol. 7, pp. 308–315, Apr. 2005.

[41] J. Schrder, S. Goetze, and J. Anemller, "Spectro-temporal Gabor filterbank features for acoustic event detection," *and Language Processing IEEE/ACM Transactions on Audio, Speech*, vol. 23, pp. 2198–2208, Dec. 2015.

[42] A. Rakotomamonjy and G. Gasso, "Histogram of gradients of time –frequency representations for audio scene classification," *and Language Processing IEEE/ACM Transactions on Audio, Speech*, vol. 23, pp. 142–153, Jan. 2015.

[43] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[44] B. Elizalde, H. Lei, G. Friedland, and N. Peters, "An i-vector based approach for audio scene detection," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, 2013.

[45] R. E. Schapire, "The boosting approach to machine learning: An overview," in *Nonlinear estimation and classification*, pp. 149–171, Springer, 2003.

[46] D. Li, J. Tam, and D. Toub, "Auditory scene classification using machine learning techniques," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, 2013.

[47] Y. Petetin, C. Laroche, and A. Mayoue, "Deep neural networks for audio scene recognition," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*, pp. 125–129, IEEE, 2015.

[48] J. L. Dai Wei, P. Pham, S. Das, and S. Qu, "Acoustic scene recognition with deep neural networks (dcase challenge 2016),"

[49] S. H. Bae, I. Choi, and N. S. Kim, "Acoustic scene classification using parallel combination of lstm and cnn," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, 2016.

[50] O. Dikmen and A. Mesaros, "Sound event detection using non-negative dictionaries learned from annotated overlapping events," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*, pp. 1–4, IEEE, 2013.

[51] A. Temko and C. Nadeu, "Acoustic event detection in meeting-room environments," *Pattern Recognition Letters*, vol. 30, no. 14, pp. 1281–1288, 2009.

[52] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Random regression forests for acoustic event detection and classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 20–31, 2015.

[53] J. Li, W. Dai, F. Metze, S. Qu, and S. Das, "A comparison of deep learning methods for environmental sound," *arXiv preprint arXiv:1703.06902*, 2017.

[54] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, pp. 16–34, May 2015.

[55] Z. Ghahramani and M. I. Jordan, "Factorial hidden markov models," in *Advances in Neural Information Processing Systems*, pp. 472–478, 1996.

[56] M. Kantardzic, *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons, 2011.

[57] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 463–484, July 2012.

[58] C. Joder, S. Essid, and G. Richard, "Temporal integration for audio classification with application to musical instrument classification," *and Language Processing IEEE Transactions on Audio, Speech*, vol. 17, pp. 174–186, Jan. 2009.

[59] J. G. Wilpon, C. H. Lee, and L. R. Rabiner, "Improvements in connected digit recognition using higher order spectral and energy features," in *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*, pp. 349–352 vol.1, Apr 1991.

[60] H. Hermansky and S. Sharma, "Traps-classifiers of temporal patterns.," in *ICSLP*, pp. 1003–1006, 1998.

[61] A. Meng, P. Ahrendt, J. Larsen, and L. K. Hansen, "Temporal feature integration for music genre classification," *and Language Processing IEEE Transactions on Audio, Speech*, vol. 15, pp. 1654–1664, July 2007.

[62] S. Lucey, T. Chen, S. Sridharan, and V. Chandran, "Integration strategies for audio-visual speech processing: applied to text-dependent speaker recognition," *IEEE Transactions on Multimedia*, vol. 7, pp. 495–506, June 2005.

[63] G. Roma, W. Nogueira, P. Herrera, and R. de Boronat, "Recurrence quantification analysis features for auditory scene classification," *IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep*, 2013.

[64] D. Chakrabarty and M. Elhilali, "Exploring the role of temporal dynamics in acoustic scene classification," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 1–5, Oct. 2015.

[65] Y. M. Costa, L. Oliveira, A. L. Koerich, F. Gouyon, and J. Martins, "Music genre classification using LBP textural features," *Signal Processing*, vol. 92, no. 11, pp. 2723–2737, 2012.

[66] W. Yang and S. Krishnan, "Combining temporal features by local binary pattern for acoustic scene classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 1315–1321, June 2017.

[67] M. Xu, L.-Y. Duan, J. Cai, L.-T. Chia, C. Xu, and Q. Tian, "Hmm-based audio keyword generation," in *Pacific-Rim Conference on Multimedia*, pp. 566–574, Springer, 2004.

[68] I. A. McCowan and S. Sridharan, "Multi-channel sub-band speech recognition," *EURASIP Journal on Advances in Signal Processing*, vol. 2001, no. 1, p. 569196, 2001.

[69] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Proc. 12th IAPR Int Pattern Recognition Vol. 1 - Conf. A: Computer Vision amp; Image Processing. Conf*, vol. 1, pp. 582–585 vol.1, Oct. 1994.

[70] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 971–987, July 2002.

[71] G. Giacinto and F. Roli, "An approach to the automatic design of multiple classifier systems," *Pattern recognition letters*, vol. 22, no. 1, pp. 25–33, 2001.

[72] A. Mesaros, T. Heittola, T. Virtanen, E. Fagerlund, A. Hiltunen, and T. Heittola, "Tut acoustic scenes 2016, development dataset," 2016.

[73] Y. E. Kim, D. S. Williamson, and S. Pilli, "Towards quantifying the" album effect" in artist identification.," in *ISMIR*, pp. 393–394, 2006.

[74] A. Mesaros, T. Heittola, and T. Virtanen, "Tut database for acoustic scene classification and sound event detection," in *In 24rd European Signal Processing Conference 2016 (EUSIPCO 2016)*, 2016.

[75] M. Kleinschmidt, "Methods for capturing spectro-temporal modulations in automatic speech recognition," *Acta Acustica united with Acustica*, vol. 88, no. 3, pp. 416–422, 2002.

[76] C. Nadeu, D. Macho, and J. Hernando, "Time and frequency filtering of filter-bank energies for robust hmm speech recognition," *Speech Communication*, vol. 34, no. 1, pp. 93–114, 2001.

[77] P. Somervuo, "Experiments with linear and nonlinear feature transformations in hmm based phone recognition," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 1, pp. I–52, IEEE, 2003.

[78] P. Jain and H. Hermansky, "Beyond a single critical-band in trap based asr.," in *INTERSPEECH*, 2003.

[79] M. Kleinschmidt, "Localized spectro-temporal features for automatic speech recognition.," in *Interspeech*, Citeseer, 2003.

[80] T. Ezzat and T. A. Poggio, "Discriminative word-spotting using ordered spectro-temporal patch features.," in *SAPA@ INTERSPEECH*, pp. 35–40, Citeseer, 2008.

[81] S. Ntalampiras, I. Potamitis, N. Fakotakis, and S. Kouzoupis, "Automatic recognition of an unknown and time-varying number of simultaneous environmental sound sources," *World Academy of Science, Engineering and Technology*, vol. 59, pp. 2097–2101, 2011.

[82] 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.

[83] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation," *Proc CHiME*, pp. 36–40, 2011.

[84] A. Dessein, A. Cont, and G. Lemaitre, "Real-time detection of overlapping sound events with non-negative matrix factorization," in *Matrix Information Geometry*, pp. 341–371, Springer, 2013.

[85] A. S. Bregnian, "Auditory scene analysis: Hearing in complex environments," 1993.

[86] R. D. Patterson, K. Robinson, J. Holdsworth, D. McKeown, C. Zhang, and M. Allerhand, "Complex sounds and auditory images," *Auditory physiology and perception*, vol. 83, pp. 429–446, 1992.

[87] B. R. Glasberg and B. C. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing research*, vol. 47, no. 1, pp. 103–138, 1990.

[88] N. Ma, "An efficient implementation of gammatone filters," 2012.

[89] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[90] C. V. Cotton and D. P. W. Ellis, "Spectral vs. spectro-temporal features for acoustic event detection," in *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 69–72, Oct 2011.

[91] M. N. Schmidt and M. Mørup, *Nonnegative Matrix Factor 2-D Deconvolution for Blind Single Channel Source Separation*, pp. 700–707. Springer Berlin Heidelberg, 2006.

[92] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, pp. 556–562, 2001.

[93] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs," *Independent Component Analysis and Blind Signal Separation*, pp. 494–499, 2004.

[94] R. E. Schapire, *Explaining AdaBoost*, pp. 37–52. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[95] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.

[96] G. E. Poliner and D. P. W. Ellis, "A discriminative model for polyphonic piano transcription," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, p. 048317, 2006.