OVERLAPPED SPEECH AND MUSIC SEGMENTATION USING SINGULAR SPECTRUM ANALYSIS AND RANDOM FORESTS

DURAID YEHYA MOHAMMED

School of Computing, Science and Engineering

University of Salford, Manchester, UK

Submitted in Partial Fulfilment of the Requirement of

the Degree of Doctor of Philosophy

September 2017

In the name of Allah, the Gracious, the Merciful.

Alif, Lam, Meem. This is a perfect Book; there is no doubt in it. it is a guidance for the righteous; Who believe in the unseen and observe Prayer, and spend out of what We (Allah) have provided for them; and who believe in that which has been revealed to thee [O Muhammad], and that which was revealed before you (all other prophets), and they have firm faith in the Hereafter. Those are upon [right] guidance from their Lord, and it is those who are the successful. (Holy Quran Chapter 2 (Surat Al-Baqara): verses 1:5)

Meanwhile in writing, there is always going to be something that can be improved or you will catch a mistake in the second, third, or even in the tenth run through, and as Abdul Raheem Albasanee Quote in the (12th Century):

"I have not seen in this day and age, that a human writes a book in which he sees no deficiency. He always thinks that if I added such and such or I removed such and such it will be better. This is an indication to the limitation of the human brain to be perfect in articulation the first time."

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude towards our God (Allah), who is the all-powerful and all-knowing creator; Allah has bestowed countless blessings upon us. Some examples of which are, Allah has endowed us with the gifts of sight and hearing, the intellect, health, wealth and family. Allah has even subjected everything in the universe for us: the sun, the moon, the heavens and the earth, and many countless things, as the Qur'an states, "If you tried to number Allah's blessings, you could never count them." (Holy Quran, Surat Al-Maa'idah, 16:18).

Secondly, this thesis epitomises not only my struggle at the writing up, it is an achievement in more than three years of effort at Salford University and precisely within my supervisors. My experience at audio signal processing has been nothing less than acceptable. Since my PhD study starting on October 1st, 2013 I have felt that I have been given unique chances to work and learn from them and start taken advantage of them. From the beginning to the end of these years, I have learned to make my interest and passion in cutting edge research. This study displays the lessons learned in investigate one of those cutting edge research: this thesis is the result of work by me with my supervisors, who I wish to thank them. Primarily, I wish to thank my consultant, Dr. Francis Li, my supervisor. He has been inspirational since the first days I started functioning of the origins tested using the tried and tested method; I remember he used to say something like "you're a good researcher and have a good programming skills and need for more planning and scientific communication!" to encourage me to contact and learn more from the experience of other researcher and follow the cleverer research approach. Ever after, Francis has assisted me not just by offering research supervision, but also academically and directed me over almost a year of development through the coarse

road to end this study. Thanks to him, I had the chance to enrich my knowledge. In conjunction, I wish to thank and highly appreciate Dr. Philip Duncan, my co-supervisor. He was the man who saved the teamwork, on every hitch and obstacle of every dilemma or challenge throughout the study journey by means of his professional knowledge in principles of the digital signal processing and by keep me focusing on the contribution. He supports me academically and emotionally to crop up with the research theme. Meanwhile, at the most difficult times when writing this thesis, he gave me the proper support and the freedom I wanted to progress on. I acknowledge that he provide me with all support as an expert in digital signal processing. My thesis team or committee directed and guided me over all the research time and work. Thankfulness to both of them for being my foremost mentors.

Also indigent of acknowledgement that I am very grateful to my PhD studentship sponsorship by the Iraqi government representative by "the Ministry of Higher Education and Scientific Research" which makes this study in this field possible and helped provide such an educational environment to work in. Furthermore, I would also like to recognise with much thankfulness the role of all those employees involved in Iraqi Cultural Attaché in London throughout my PhD studies, notably the former and present Counsellor Attaché (Professor Musa al-Musawi and Professor Hassan Al Alak).

An exceptional appreciation ought to also be given to my fellow post-graduates (Anugrah Sabdono Sudarsono (Nanoo), Alex Wilson, James F Massaglia, Usman Ali, Joshua Meggitt, Ahmed Al-Noori, Will Bailey, Omar Eldwaik and Khamis Ahmed and Nikhilesh Patil) in the room G10/11 and the new postgraduate office for the countless enjoyable conversions, snacks and times.

Outside the sphere of the Newton building there are a few people whose support and friendship I would like to acknowledge; all my brothers, all my friends back home. Also, I feel this acknowledgement would not be complete without recognition of Muhammed M. Al-Maathidi, who introduced me to Francis Li and set me on this path. I also wish to express sincere thanks to my home University, Iraqi University College of Education for Women, Baghdad, and colleagues for their help and support.

On a more personal note, I thank with love Amel and Abdulrahman, my wife and son. Considering me most as a Ph.D candidate, Amel herself has been my best friend and great companion, loved, supported, encouraged, entertained wife and helped me get through this sustained challenging period of time in the most positive way. Again, I need to write something about my new baby "AFNAN", who has only seven days old meantime my writing up to this part from the thesis. Also, I would like to express my emotional demands that come with caring for a new-born baby and helped me continue through this distressing period.

Last but not the least, I must thank my parents, who they passed away; and I can't say more than Qur'an states, [And lower to them the wing of humility out of mercy and say, "My Lord! Have mercy on them both as they did care for me when I was little"] (Holy Quran, Surat Al-isrā, 17).

Without their care, consultation, consolation, encouragement and..etc. I would absolutely not be in the position which I am in today, and for that I am eternally thankful.

LIST OF ABBREVIATIONS AND ACRONYMS

AASP	Audio and Acoustics Signal Processing challenge
ASR	Automated Speech Recognition
CHIL	Computers in the Human Interaction Loop
dBs	decibels
DT	Decision Tree
DFT	Discrete Fourier Transform
EEG	Electroencephalography
Entrocy	ENTROpy frequenCY
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
MARSYAS	Music Analysis, Retrieval and Synthesis for Audio Signals
MCR	Mean Crossing Ratio
MCR	Mean Crossing Ratio
MFCC	Mel-frequency Cepstral Coefficients
MIR	Music Information Retrieval
NMF	Non-negative Matrix Factorization
OL	Office Live
PCs	Principal Components
pmf	Probability mass function
RFs	Random Forests
RMS	Root Mean Square
SC	Spectral Centroid
SNR	Signal to Noise Ratio
SS	Spectral Spread
SSA	Singular Spectrum Analysis
std	Standard deviation
SVM	Support Vector Machine
UAR	Unweighted Average Recall
VAD	Voice Activity Detection
ZCR	Zero Crossing Rates

SYMBOLS

Overall annotations: notations and equations are for the most part given in the time domain, except as otherwise specified. For simplicity and clarity, signals and frames in the time domain are shown in italics and lowercase, whilst, frequency domain notations are denoted in italic uppercase. Vector quantities are shown in bold lowercase whilst matrix quantities are indicated by bold uppercase font. Individual matrix and vector elements are shown in an italic typeface and their position indicated by subscripts i, j. Symbol definitions are tabulated below. Symbols used only in passing are not included.

N_t	The total length of the given audio signal in a number of samples.	*S(k)*	The audio spectrum calculated for the audio frame.
L	The length of the frame (the total number of samples in each frame)	*P(k)*	The squared magnitude audio spectrum estimated for the *S(k)* of the audio frame.
hop	The hop size in samples (the size of window shifting measured in samples).	*k*	The bin index of the frequency.
s(n)	Audio signal	N_{FT}	The size of the spectrum.
n	Represents the time index of the audio signal samples	$\wedge$	Estimated value
f(x)	Audio Frame	**U**	Eigenvectors matrix
n	The time index of the audio frame sample.	**e**	Eigenvector
i	The index of the frame position ($1 \leq i \leq L$)	μ	The statistical Average
fe	Audio Feature	φ	The Signal phase
arg max	Maximum value in the vector	**Cx**	Covariance Matrix
arg min	Minimum value in the vector	*m*	Number of training samples
NF	The total number of frames in the audio signal *s(n)*.	*Nof*	Number of calculated features
Fs	Sampling rate	λ	Eigenvalue
$\forall$	Denotes for all	Λ	Lambda
R	Real number	*K*	Number of embedding vectors
		L_w	Length of embedding vectors

ABSTRACT

Recent years have seen ever-increasing volumes of digital media archives and an enormous amount of user-contributed content. As demand for indexing and searching these resources has increased, and new technologies such as multimedia content management systems, enhanced digital broadcasting, and semantic web have emerged, audio information mining and automated metadata generation have received much attention. Manual indexing and metadata tagging are time-consuming and subject to the biases of individual workers. An automated architecture able to extract information from audio signals, generate content-related text descriptors or metadata, and enable further information mining and searching would be a tangible and valuable solution.

In the field of audio classification, audio signals may be broadly divided into speech or music. Most studies, however, neglect the fact that real audio soundtracks may have either speech or music, or a combination of the two, and this is considered the major hurdle to achieving high performance in automatic audio classification, since overlapping can contaminate relevant characteristics and features, causing incorrect classification or information loss.

This research undertakes an extensive review of the state of the art by outlining the well-established audio features and machine learning techniques that have been applied in a broad range of audio segmentation and recognition areas. Audio classification systems and the suggested solutions for the mixed soundtracks problem are presented. The suggested solutions can be listed as follows: developing augmented and modified features for recognising audio classes even in the presence of overlaps between them; robust segmentation of a given overlapped soundtrack stream depends on an innovative method of audio decomposition using Singular Spectrum Analysis (SSA) that has been studied extensively and has received increasing attention in the past two decades as a time series decomposition method with many applications; adoption and development

of driven classification methods; and finally a technique for continuous time series tasks.

In this study, SSA has been investigated and found to be an efficient way to discriminate speech/music in mixed soundtracks by two different methods, each of which has been developed and validated in this research. The first method serves to mitigate the overlapping ratio between speech and music in the mixed soundtracks by generating two new soundtracks with a lower level of overlapping. Next, feature space is calculated for the output audio streams, and these are classified using random forests into either speech or music. One of the distinct characteristics of this method is the separation of the speech/music key features that lead to improve the classification performance.

Nevertheless, that did encounter a few obstructions, including excessively long processing time, increased storage requirements (each frame symbolised by two outputs), and this all leads to greater computational load than previously. Meanwhile, the second method employs the SSA technique to decompose a given audio signal into a series of Principal Components (PCs), where each PC corresponds to a particular pattern of oscillation. Then, the transformed well-established feature is measured for each PC in order to classify it into either speech or music based on the baseline classification system using a RF machine learning technique. The classification performance of real-world soundtracks is effectively improved, which is demonstrated by comparing speech/music recognition using conventional classification methods and the proposed SSA method. The second proposed and developed method can detect pure speech, pure music, and mix with a much lower complexity level.

1 INTRODUCTION

1.1 Introduction

Since the invention of the audio recorder in 1877, of the motion picture camera in 1880, and of the video recorder in 1951, a great many archives of recorded soundtracks have come into existence. Over and above this, recent years have seen the archiving of ever-increasing volumes of digital media, along with an enormous amount of user-contributed content. Digitisation can preserve the content; however, the usability of the data is limited if there is no tangible way to search for interesting content from the mass of data. Such metadata is essential for semantic analysis, indexing, searching, and many other applications. Manual annotation methods for metadata, i.e. data about the content, are time-consuming, prone to errors and sometimes biased. Consequently, there is a pressing need for automated classification, recognition, and information mining of audio content.

There is a considerable body of research that promotes audio content analysis through recognition of particular audio classes (speech or music) in a mutually exclusive manner and under specific conditions (e.g. Khonglah and Prasanna, 2016, Eyben, 2016 and Khaldi et al., 2016). To extract keywords or semantic meaning from soundtracks, techniques such as speech recognition, music information retrieval, and event sound detection can be employed. Therefore, audio classification is a key pre-processing stage for automated semantic audio content analysis and metadata generation.

Automatic classification of real-world audio soundtracks into speech and/or music, when the two sometimes overlap, is a particularly challenging problem. Although it is well understood that the overlapping of the classes might mitigate the information re-

trieval system's performance, little research has been performed with the aim of addressing this problem (e.g. Lee and Ellis, 2008, Sell and Clark, 2014, Tomonori et al., 2008). Zhang and Kuo (2001) presented an approach to annotation based on segmentation and annotation of audio data into three main categories, namely components comprising silence, those with music, and those without music. Segments falling into each of the last two categories are then further classified into more components. The Audio and Acoustics Signal Processing challenge (AASP) (Giannoulis et al., 2013), which is sponsored by the IEEE Signal Processing Society, is a worldwide competition of technical innovation to classify real-world scenario signals with and without overlapping issues. The scope is limited to speech and other indoor/outdoor events; no music has been included, and its main objective is to detect prominent events and to ignore the remaining content of the soundtracks. One conclusion drawn from the AASP was that "the task of recognising individual potentially overlapping sounds becomes significantly challenging and the performance of systems that are even prepared to deal with polyphonic content falls dramatically" (Giannoulis et al., 2013). Although the AASP is about speech and event sounds, the conclusion from this large-scale competition suggests that the technically challenge in handling overlapped audio classes remains unsolved.

In fact, the literature on audio content analysis has concentrated principally on classical classification, i.e. categories are logically exclusive, such that an element is assumed to be a member of one class and of that class only, such as speech/music discrimination in

the non-overlap condition. This hinders some attempts to put these techniques to practical use in audio information mining since a segment of the soundtrack can have either speech, music, event sounds or any combination thereof, as shown in Figure 1-1.

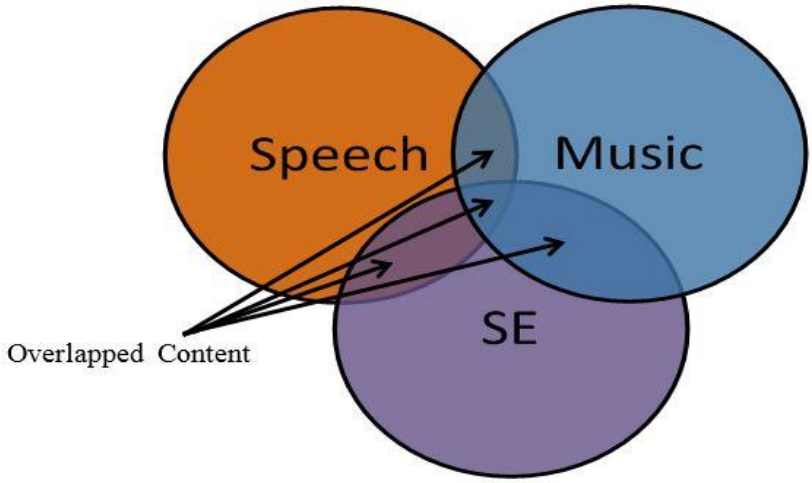


Figure 1-1 The contents of overlapped soundtracks

There are several effective audio feature and machine learning techniques which have reported satisfactory performance in recognising a particular class in the controlled condition. However, the overlapped nature of audio content represents the greatest challenge for information retrieval systems since it can contaminate the characteristics and features of the overlapped classes such that they cannot be classified correctly, with a classical classification method, without losing useful information. In Duncan et al. (2014) and Mohammed et al. (2015), a new technique has been proposed for non-exclusive classification through using a timestamp with three classifiers, each of which functions as a sensor to detect its respective class even when overlapping takes place. Consequently, the start and end of each class can be determined even when overlapped.

Speech/music cleaning or enhancement algorithms have been suggested as a possible solution, and several techniques have been proposed in the literature, even though there is no software available in the public domain - neither commercial nor free - able to

accomplish speech/music separation with mono channel recording, giving adequate performance without prior knowledge about a given signal. If knowledge of the recorded voice signal is available, this can help in developing specific spectral masking/subtraction techniques allowing separation of the spectral content of the recorded voice in the time-frequency representation of the mixed signal. However, since real sounds are composed of a comb-pattern of harmonics exponentially spaced in frequency, it remains almost impossible to resolve the harmonic overlap problem in the mixed signal, which means that a distortion will inevitably be introduced into the separated signals after processing. Consequently, mitigating the overlapping between components through separating them into a number of oscillations with a lower ratio of overlapping and then classifying them separately is key.

The overall purpose of this study is to investigate the potential of singular spectrum analysis (SSA), which represents an analysis of time series utilising the singular spectrum as an efficient decomposition tool to improve the classification of arbitrary soundtracks. It is worth noting here that, with the overlapped nature of the audio content, in order to correctly determine the target class a new criterion is required. On the other hand, it is not the purpose of this study to separate the audio sources nor enhance the components, but rather to determine the contents of each segment of the audio soundtrack. Due to there being a wide range of kinds of acoustic events, with unlimited characteristics, in the world, the scope of the present thesis is limited to speech and music.

The SSA algorithm is used for the classification of mixed soundtracks. The first method is to adapt existing methods by reducing the degree of overlap between different classes of audio content, in order to mitigate classification difficulties and improve the performance of automatic classification. The second method is to develop an algorithm for

mixed soundtrack classification using transformed features extracted using SSA. Furthermore, an augmented and modified feature set has been developed which has the ability to detect music even when overlapping takes place. This generates a significant feature set for mixed soundtrack classification. Comparison of classification accuracy, specificity, and sensitivity against those of another state of the art method is made. The performance shows promise.

Classification accuracy was also compared with existing baseline classification methods; the developed algorithm has outperformed other methods based on results published in the literature.

1.2 The Aim and Objectives of the Study

The aim of this study is to develop an automatic classification of overlapped soundtracks using an SSA algorithm. In other words, the aim is to investigate the capabilities of Singular Spectrum Analysis in overlapped soundtrack classification. "Can SSA be effectively used to mitigate the classification difficulties associated with overlapping between ingredients of mixed sounds and then improve the performance of automatic classification?" is the research question.

1.3 The objectives

The objectives of this study may be more specifically stated as

- To carry out a detailed literature and background study.
- To generate and prepare a suitable dataset to conduct an experimental study of the suggested methods in this thesis.
- To evaluate, interpret, and justify the existing classification method.
- To study and clarify the SSA technique as a widespread method for noise isolation.

- To develop an SSA-based novel method for overlapped soundtrack classification; this is accomplished by two different methods and through a number of processes.
- To evaluate the usability of the SSA method with regard to real worked mixed samples.
- To identify the limitations of the method developed through this study and suggest future work.

1.4 The outline of the Thesis

This thesis is organised as follows: Chapter 1 gives a general introduction. Chapter 2 presents a literature review of most of the relevant work, including the current state of the art in the audio information retrieval, in which content classification is typically used as a pre-processing stage. Chapter 3 investigates the anatomy of the feature space, the computation of the common features in the literature, and general comparisons between speech and music characteristics. Chapter 4 reviews machine-learning models for audio classification and highlights the Random Forests (RFs) and Decision Tree (DT) algorithms to be adopted in this thesis. Chapter 5 gives a general demonstration of the applied datasets and the generation of mixed soundtracks. Chapter 6 proposes a multi-iterative algorithm using MARSYAS (Music Information Retrieval and SYnthesis System) with a method employing spectral subtraction algorithms for non-exclusive classification. Chapter 7 demonstrates set of newly developed features. Chapter 8 provides a brief description of the methodology of Singular Spectrum Analysis and its application, and gives an example. Chapter 9 describes the proposed methods using the Singular Spectrum Analysis methodology for overlapped soundtrack classification. It also explains a bespoke method, parameters, decomposition, and classification for PCs corresponding to soundtracks through transformed features calculation. Chapter 10 reports the results of this research combined with a discussion thereof. Chapter 11 includes an

explanation of the usability of the proposed methods with SSA with real-worked soundtracks. Finally, Chapter 12 summarizes the present work, draws the conclusions and limitations of this study, and suggests future work.

1.5 Publications Outcome from this Research

1.5.1 Journals and Conferences Papers

1. Duncan, P., Mohammed D., and Li F., 2014. "Audio Information Mining–Pragmatic Review, Outlook, and A Universal Open Architecture" Audio Engineering Society Convention 136, Berlin, Germany, 24 April. Audio Engineering Society AES, p. 9075.
2. MOHAMMED, D., DUNCAN, P., AL-MAATHIDI, M. M. & LI, F. F. 2015. "*A System for Semantic Information Extraction from Mixed Soundtracks Deploying MARSYAS Framework"*. 13th International Conference on Industrial Informatics INDIN Cambridge, UK: IEEE.
3. Mohammed, D.Y., Duncan, P.J. and Li, F. F., "Audio information extraction from arbitrary sound recordings*"*, in: 22nd International Congress on Sound and Vibration (ICSV22), 12th - 16th July, Florence, Italy.
4. Mohammed, D.Y., Duncan, P.J. and Li, FF, "*Audio Content Analysis in The Presence of Overlapped Classes- A Non-Exclusive Segmentation Approach to Mitigate Information Losses"*, Global Summit and Expo on Multimedia & Applications August 10-11, 2015 Birmingham, West Midlands, UK.
5. Mohammed, D. Y., Li, F. F., "*Overlapped Soundtracks Segmentation: a Singular Spectrum Analysis and Random Forests Approach*", Accepted in IEEE ICKEA 2017 Conference (under publishing stage).

6. Mohammed, Duraid, Li, Francis; "*Overlapped Soundtracks Classification Using Singular Spectrum Analysis and Random Forests*", submitted to IEEE Access journal 12th-Jun-2017.
7. Mohammed et al.., 2016. "*The Extraction of Semantic information from Arbitrary Audio Soundtracks Recording*", Proceedings of the CSE 2016 Annual PGR Symposium (CSE-PGSym 16).

1.5.2 Posters in Published Conferences Proceeding

1. MOHAMMED, D. Y., DUNCAN, P. J.& LI, F. F. 18th June 2014, Poster-Dean Showcase, Salford University, Manchester, Uk, "*Audio information Mining, Programmatic review*" and won the prize for the best poster.
2. MOHAMMED, D. Y., DUNCAN, P. J.& LI, F. F. 2015 , "*Audio information Mining for arbitrary soundtracks recordings*", Poster-Dean Showcase, Salford University, Manchester, Uk, 28 June 2015.
3. Furthermore, Build the suggested UOA (Universal Open Architecture) system toolbox and participate in the university fellowship. The toolbox won the second round of Innovation fellowship.
4. MOHAMMED, D. Y., DUNCAN, P. J. & LI, F. F., 2016, "*A System for Semantic Information Extraction from Mixed Soundtracks Deploying MARSYAS Framework", AES UK Graduate Student Poster Competition, Oxford on 26th February 2016.*

2 LITERATURE REVIEW AND BACKGROUND OF AUDIO CLASSIFICATION SYSTEM

2.1 Introduction

In the present chapter, the current state of the art in the field of Audio Content Classification is highlighted. There are many papers related to this field. Exhaustive review of all the related papers is not the intention, only some major milestones and those upon which subsequent work in the thesis is built are listed. The chapter is organised as follows. In section 2.2 work related to this study is discussed. Section 2.3 presents a literature review from the point of view of audio classification architecture including preprocessing steps (framing, frame size selection, and windowing), feature extraction and machine learning techniques.

2.2 Audio Classification Systems

Audio classification systems analyse the given audio signal and generate labels that describe the signal. These labels are used to characterise the target class's segments in accordance with the classification system. Figure 2-1 exhibits the common architecture of the classification systems.

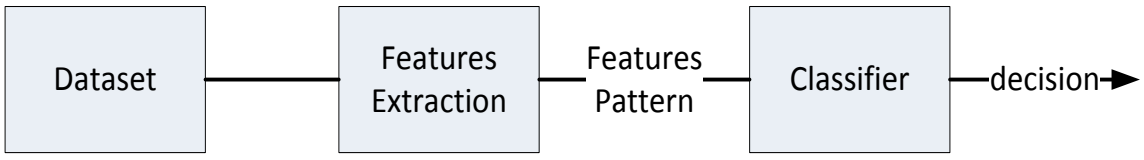


Figure 2-1 Classification system Architecture

The categorization/classification can be done on the basis of three stages: at the preprocessing stage the input signals are sectioned into small parts called frames; then, the

classification system processes each of these frames separately to classify it into one of the classes as above. Furthermore, normalisation and windowing are significant processes for bringing the input frame to the normative of the classification form; this will be explained in more detail later on in this study. Sets of time domain, frequency domain, and time-frequency domain features are extracted and used for data reduction and characterization of audio content. The most common and successful features, identified through the literature review, are selected in this study for real-world audio classification. In addition, a modified feature is developed to complement the other features in the classification process. The final stage is the machine learning technique, which is used to tackle recognition or classification problems with audio samples (Bishop, 2006). The common aim of pattern recognition algorithms is learning and generalisation. In other words, the classifier acquires certain rules from training data which are generally suitable for solving all similar problems despite their having a different dataset.

In general, audio content analysis refers to extraction and retrieval of information from audio content that depends on the extracted features. In this study, real-world audio signal classification, with some emphasis on the overlapping of the speech/music classes, has been carried out.

The adoption and development of classification methods are typically application-driven. During the past decades, many bespoke segmentation and classification systems have been implemented; Table (2-1) lists a number of bespoke systems for audio classification.

Table 2-1 Summary of Past Applications

Application	Role	Data	Deployed Features	Limitations
Program classifier (Dhanalakshmi et al., 2009)	Classify radio or TV channels	Programmes are classified into six different categories (music, news, cartoon, movie, sports and advertisements)	- Linear prediction analysis - Mel-frequency cepstral coefficients	Works with the broadcast channel. Classifies into categories, not classes.
Audio clustering (Lu et al., 2001a)	Classify audio into 5 clusters	- Silence - Music background sound - Pure speech - Non-pure speech - Speech over noise or music.	- ZCR - Time Energy. - Spectrum Flux. - Linear Predictive Coefficient. - Band Periodicity. - Noise Frame Ratio	Exclusive clustering. Authors' own data.
Speech, music discrimination (Saunders, 1996)	Discrimination between Speech and music.	- News channel	- Tonality. - Bandwidth. - pitch. - Tonal duration. - Energy	Works on broadcast news channel only.
Football game referee (Lefèvre and Vincent, 2011)	Whistle sounder detection	Football games		Detects specific sound.
Surveillance systems (Meinedo and Neto, 2003)	Detect specific audio events	Predetermined events, or sound level.		Unable to recognise sound types or classes.
MARSYAS [software] (Tzanetakis, 2014)	Music Synthesis	Music and Music genres Speech		Exclusive audio classification
CHIL (Waibel1 et al., 2004)	Events Detection	Who and where ‘what’ and ’why and how.'		Predetermine events only (in office or lecture room)
Dragon [software]	Speech Recognition	Speech signal		Signal with high SNR is required

2.2.1 Framing

Traditionally, the audio file is partitioned using a time window into a series of consecutive analytical frames of limited length, with or without overlapping. Many research studies have shown that, in general, frame sizes in the range 10-40 ms are appropriate, some examples of which are Huang and Hansen (2006), Wang et al. (2000), Kim et al. (2005). Others, such as Saunders (1996), Scheirer and Slaney (1997) have used a longer window with fixed size up to 2.4 s, and their research achieved a high accuracy rate when discriminating speech from music on a broadcast news channel. In fact, this makes it possible to look at state transition processes over consecutive frames. In particular, each 2.4 s window (or segment) contained 150 non-overlapped frames. The experimental results gave a 98% success rate (Scheirer and Slaney, 1997), distinguishing music from speech with an error rate of 5.8% when using frame by frame parsing and 1.4% when the window size of 2.4s was used.

Lu et al. (2001b) provide a different scenario for audio classification, classifying audio through its contents into five classes (silence, music background sound, pure speech and non-pure speech (speech over noise or music)) by Support Vector Machine (SVM). The audio file was segmented into one-second segments, and each of these segments was subdivided into non-overlapping frames of 25 ms before classifying these segments into one of the above five classes. This work depended on the extraction of two sets of features. The first set consists of Mel Frequency Cepstrum Coefficients (MFCCs) and the second set comprises perceptual features including short time energy, Zero Crossing Rates (ZCR), sub-band power distribution, brightness, bandwidth and pitched ratio (ratio between the number of pitched frames and the total number of frames in a sub-clip), spectrum flux, linear spectrum pair, and band periodicity. In addition to these two sets of feature spaces, the mean and standard deviation of the feature spaces across 40

frames within each 1-second segment were also computed and used. Energy information is employed at the beginning of the algorithm to discriminate between silence and non-silence. At the next stage the non-silence is further divided into the last four classes, which are detected using 3 SVMs as follows: SVM1 divides non-silence into with speech and without speech; SVM2 splits the speech clip into non-pure speech and pure speech; and SVM3 divides without speech clips into background sounds and music.

This seems to suggest that zooming in signal processing and analysis, and the combined use of both short-term analytical frames and slightly longer-term segments are particularly beneficial. Typically, if the frame length is too short then it will not have enough samples to obtain reliable spectral information, and if it is much longer, then the signal will change significantly throughout the frame.

Also, the window function is used to reduce the edge effect of the framing. There are a number of window types, one of the most popular being the Hanning window, which is used for general purpose applications because of its low spectral leakage (low distortion and the ability to recover the original signal from the converted one is high), with the trade-off being a slightly decreased resolution (widening of the main lobe) (Harris, 1978). In this work, the Hanning window has been used as a filter window to reduce the edge discontinuities (Kim et al., 2005).

2.2.2 Feature Extraction

Feature extraction involves processing the segmented audio data to generate statistically significant observations and other salient information, and is essentially a form of data reduction. Commonly used features are frequency domain and time domain statistics. Time-frequency domain approaches are becoming increasingly popular (Kos et al., 2013, Webb, 2011).

A survey of comprehensive research concerning feature extraction can be found in Shao et al., 2003 and MITROVIC et al., 2010, including taxonomy tables of more than 73 parameters that can effectively represent audio features in a diverse variety of ways.

Features are usually extracted from overlapped frames (each frame includes a number of samples). The section above lists a number of researchers who tested different frame sizes in their research; they show the effects of the frame's size on the results. Giannakopoulos (2009) extracted a set of raw and statistical features from 100 ms frames without overlapping for harm detection (violent sound detection e.g. gun sound, screaming and so on) in audio content. The final decision is made on the basis of the statistical features (standard deviation, mean, median).

For speech/music discrimination, Sell and Clark (2014) derived new features from the Chroma vector based on the musical tonality. Kos et al. (2013) proposed a new set of features (Energy Variance of Filter Bank) for speech/non-speech segmentation. Kos mentioned that the newly developed set of features could be deployed as a discriminator between speech and music with efficient results. Gaussian Mixture Model GMM has been used for evaluation of proposed features. For the same classification purpose, Khonglah and Prasanna (2016) investigate the behaviour of two different feature sets. The first set is related to the excitation source and contains the normalised autocorrelation peak strength of zero frequency filtered signal and the peak-to-side lobe ratio of the Hilbert envelope of the linear prediction residual. The second set denotes the vocal tract system and syllabic rate of speech and includes the log mel energy feature, which represents the vocal tract information. The modulation spectrum represents the slowly varying temporal envelope. Khonglah states that the proposed sets of features provide additional improvements in speech/ music discrimination when combined with existing features. Both the Gaussian mixture models and the support vector machines were

deployed for evaluation purposes.

Tzanetakis and Cook (1999a) implemented a real-time speech/music discrimination system with some emphasis on music retrieval, called MARSYAS (Music Analysis, Retrieval and SYnthesis for Audio Signals). The MARSYAS system provides high accuracy through applying several features (Spectral centroid, Spectral flux, Pitch, MFCC, Zero crossings Rate (ZCR), Root Mean Square (RMS), Spectral roll-off) with some statistical computations. The comparison is made between two groups of features for classifying audio files into the following audio classes (speech, noise, crowd noise, and music genres (popular music, classical music (Jazz, Folk, Electronica, Rand, Rock, Reggae, and Vocal)) (McKinney and Breebaart, 2003). The first of these contains low order signal characteristics (Root Mean Square RMS, Zero Crossing Rate (ZCR), bandwidth, spectral centroid and energy, and MFCC). The second group represents roughness, sharpness, loudness, and temporal envelope fluctuations. Gaussian-based quadratic discrimination analysis is used for classifying, and results in the evaluation. The author indicated that results could be improved by enhancements of the feature space.

Chapter Three will cover the most common audio features which have been used in the field of audio classification and deployed in the work described in the above literature.

2.2.3 Machine Learning

Machine Learning refers to an artificial process that optimises a feature extraction stage to partition the data into relevant classes. There are two main methods of classification, namely: unsupervised classification (clustering); and supervised classification (discrimination). These two have been applied to a diverse range of work including physics, mathematics, statistics, engineering, artificial intelligence, computer science, and the social sciences; see Webb (2011) for more information.

An important and growing body of literature has investigated various machine-learning techniques in the field of audio content analysis. Dhanalakshmi et al. (2009) categorized audio using Radial Basis Function Neural Networks (RBFNN), which are based on Radial Basis Function (distance function) as the activation function for hidden layer neurons and SVM reliant on event type to categorise audio into six different categories (music, news, cartoon, movie, sports, and advertisements), with accuracy of 92%. Linear predictive coefficients, linear predictive cepstrum coefficients, and Mel-frequency Cepstrum Coefficients (MFCC) were extracted from audio. SVM and neural networks were also applied with the same features to compare the result with RBFNN method. The accuracy rates were 92% and 93% respectively.

Lu et al. (2002) classified audio specifically into four categories: speech, music, environmental sounds, and silence. This was achieved using K-nearest-neighbour as the machine learning technique to discriminate the audio into classes. The speech segments were further divided into groups to denote different speakers through a developed unsupervised segmentation/classification algorithm.

Lefèvre and Vincent (2011) combined Hidden Markov Model with K-mean classifier in order to classify football games into three classes (whistle, crowd and speaker's voice). Each audio class was detected via several features. These features are computed either from a complete audio segment or from a frame (set of samples). The segment length was static at 1 second and the frame size represents 1024 samples with 512 overlapping. Lavner and Ruinskiy (2009) deployed the Decision Tree technique with time domain ZCR feature and frequency domain features such as spectral energy, MFCC, and others to discriminate between speech and music in real time audio files. The author reported that the DT algorithm outperformed SVM classifier results.

Thambi et al. (2014) used Random Forests (RFs), which is considered the modern generation of DT machine learning, to improve the discrimination between speech and non-speech to decrease the required storage space through saving only speech segments. The authors illustrated that RFs results were better than other decision tree algorithms. Also, smoothing is used over five segments to improve the results. Random forest was compared against the bagging and bootstrapping decision tree algorithms by Zhang (2015) for classifying environmental audio into five different sets (bird, wind, rain, frog, and thunder) based on various sizes of training samples; the results show the stability of random forest against the other two algorithms.

Díaz-Uriarte and Alvarez de Andrés (2006) deployed RFs for Gene selection and classification of microarray data. Due to the promising results and performance, the authors recommended that RFs become a "standard tool-box". Later on, Statnikov et al. provide a comprehensive study for the same purpose of classification between random forests and support vector machines (Statnikov et al., 2008). The author reported that RFs classifier outperformed SVM classification results.

RFs also gave an excellent output even with the small size of the training samples, due to its geometric characteristics of being treated as a collection of hyperplanes, each one orthogonal to the respective feature axis (Breiman, 2001a).

2.3 Related Work

The idea of automated media information retrieval has been around for several decades, since the 1980s when early PCs and mainframes were the predominant computing platforms. The past three decades have seen tremendous technological progress in computing power and prevalence. Alongside the computing power enabling more to be attempted, algorithms, developments in the field of Automated Speech Recognition

(ASR), Music Information Retrieval (MIR), event sound recognition, and machine audition of soundscapes have accumulated a large number of invaluable methods and tools, and much know-how. A system integration approach to audio information mining can be hypothetically built upon the successes in these fields to extract and obtain information of interest from soundtracks. Moreover, machine learning tools such as statistical and intelligent signal processing, soft computing, pattern recognition techniques and data mining methods have all been developed to an even more mature and sophisticated level. This section will present a review of the most of the relevant work and the present state of the art of these enabling and related technologies.

2.3.1 Music Information Retrieval

Studies in MIR have seen similar progress in the development of machine audition techniques. Downie has defined MIR as "a multidisciplinary research endeavour that strives to develop innovative content-based searching schemes, novel interfaces, and evolving networked delivery mechanisms in an effort to make the world's vast store of music accessible to all" (Downie, 2003). Downie has summarised the impediments against the MIR system in three simple points "No standard collection of music, no standard tasks of performance task and no standard metric".

The International MIR Systems Evaluation Laboratory (IMIRSEL) by Downie (2007) produced three projects to support the MIR area. These projects are Music Information Retrieval Evaluation Exchange (known as MIREX), which holds an annual event to evaluate the algorithms, techniques and music digital libraries in this field. Networked Environment for Music Analysis, which is abbreviated as NEMA, has planned to build an open web service framework to evaluate and investigate tools which are used in MIR as well as other applications. Structural Analysis of Large Amounts of Music Information (also called SALAMI) provides resources for musicologists for music analysis,

which has generated a database containing 23,000 hours of analysed digital music.

Figure 2-2 depicts a general MIR system. In general, all MIR systems share the following objectives:

- Automated music transcription.
- Musical genre categorization.
- Mood and theme analysis

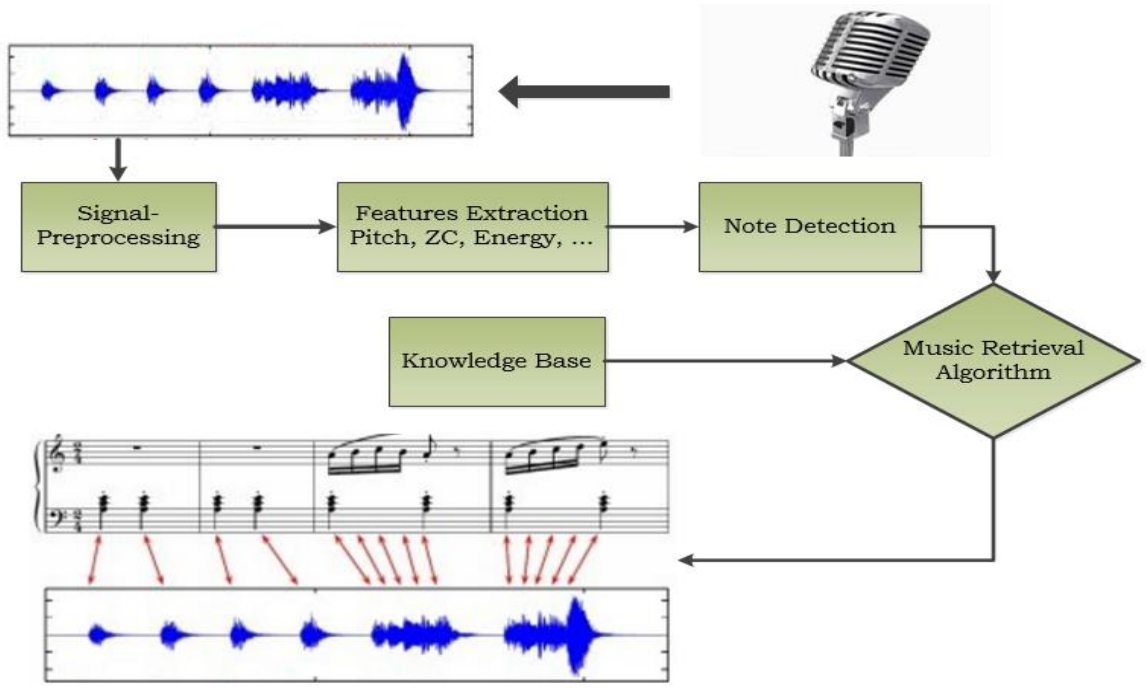


Figure 2-2 Music Information Retrieval MIR System

Table 2-2 illustrates the most prominent software and tools that have used in the field of MIR with a brief explanation of each.

Table 2-2 List including most prominent MIR software and Tools

Software	Function	Description
BeatRoot	Analysis, Music Transcription	JAVA-based beat annotation software (audio, MIDI) for beat tracking (Dixon, 2001).
Essentia	Analysis, Music Transcription	C++ library for audio analysis and audio-based music information retrieval (Bogdanov, 2013).
Humdrum Toolkit	Analysis Toolkit	Tools intended to assist in music research at the symbolic level (Huron, 1995).
jMIR	Analysis	Software/Java for automatic music classification and similarity analysis (McKay, 2010)
MIRtoolbox	Features Extraction	MIRtoolbox is a Matlab toolbox used for the computation of audio features (Lartillot, 2010)
Sonic Visualiser	Audio Analysing and features extraction	An application for viewing and analysing the contents of music audio files (Cannam, 2010)
MARSYAS	Analysis Synthesis	MARSYAS (Tzanetakis , 2009)
C++ Library for Audio and Music	C++ Programming Library	CLAM is a (C++ Library for Audio and Music) software framework for research and application development in the Audio and Music Domain (Bartkiewicz, 2013).
ChucK	Programming Language	Audio programming language (Peruse, 2015)
CLM	Synthesis	Common Lisp Music is a music synthesis and signal processing package in the Music V family (Schottstaedt, 1986).
Nyquist	Synthesis	Nyquist is a sound synthesis and composition language. Nyquist is a system based on functional programming (Chris, 1986).
SuperCollider	Environment and programming language for Features Extraction and Synthesis,	A real time audio synthesis programming language (McCartney ,1996)

MARSYAS is a widespread audio processing system with specific emphasis on MIR; it has achieved the previously mentioned objectives with some limitations such as the inability to classify non-exclusive (overlapped) soundtracks. For example, there is no ability to detect speech when it is synchronised with the music. Therefore, there is no speech transcription capability, which leads to the loss of important information. MARSYAS has been deployed by IMIRSEL as an effective evaluation tool for digital

music libraries and MIR algorithms. IMIRSEL employs this software for its ability to provide a general, extensible and flexible architecture that allows easy experimentation with algorithms and provides fast performance that is useful in developing real-time audio analysis tools.

2.3.2 Automated Speech Recognition

In the formative years of ASR, speech recognition systems processed only one word at the time. These elementary ASR systems based on template matching were effective in recognising isolated words, but not running speech. Subsequently, recognising short sentences without the need for the speaker to pause during the utterance became possible following the development of connected word detection (Noyes and Starr, 1996). Later systems proposed in Arriola and Carrasco, (1990a), Arriola and Carrasco, (1990b) applied the multi-layer perceptron and Hidden Markov Model classifiers to effectively model and predict the probabilistic nature of running speech and support much more reliable final decisions.

Khemiri et al. (2013) implemented a system of audio indexing to look for predetermined advertisements through broadcast radio. The unit called Automatic Language-Independent Speech Processing (ALISP) is used. ALISP is based on temporal decomposition and vector quantization. HMM is employed to model the system.

Commercial speech recognition systems in the last decade have developed from speaker-dependent systems, where the systems are required to adapt to individual talkers before use (Lu et al., 2002) to speaker-independent systems, which will recognise more generally. There is currently a variety of software available for ASR. Examples include Dragon, CMUSphinx, Kaldi, iATROS, VoxForge, MacSpeech, Scribe, iListen, IBM ViaVoice and google out. The University of Cambridge-Microsoft HTK toolkit

offers a set of free baseline algorithms for further development into bespoke applications (Microsoft and Cambridge, Sept. 28, 2000).

The last three decades have seen a great deal of work taking place to address ASR problems and thereby achieve much improvement in ASR systems. Nonetheless, there are still major challenges ahead, especially the robustness issue of ASRs in diverse application settings, such as language, pronunciation/intonation, signal-to-noise conditions and other related factors. Figure 2-3 shows the conceptual processing of an ASR system.

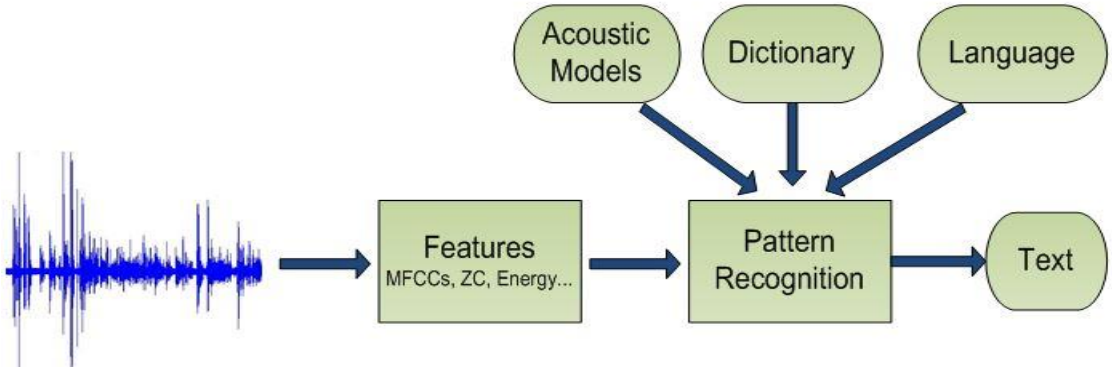


Figure 2-3 The General Architecture of Automated Speech Recognition system

2.4.3 Acoustic Event Detection

Event sounds and/or soundscapes could form a robust body of material, which might provide supplementary information for audio information mining. The information, comprising analysis of sound events, may yield a clear series consisting of a logically connected scene or events, which provides an extra dimension of information in the soundtracks.

The European Commission's integrated project by the name of CHIL ("Computers in the Human Interaction Loop") is a three-year project to analyse human face to face situations and extract knowledge in the office or lecture room (Waibel1 et al., 2004). The project scenario is based on answering a number of questions through environment

events analysis. These questions are of the type (‘who and where’, ‘what’ and ’why and how’).

Temko et al. (2006a) give an evaluation of three proposed systems for the analysis of acoustic event detection and classification of events in the meeting room. Two of the systems utilised HMM, and the other used SVM. In general, the systems classified events into speech, CHIL events, and other events. The system based on SVM gave the same results as the other two systems or better. Temko and Nadeu (2006b) also classified acoustic events in the smart meeting room “a room equipped with multiple cameras and microphones in order to investigate the video and audio perception of the computer systems” (Temko et al., 2007, pp.132), using SVM and comparing this with GMM. The SVM gave improved results over the GMM. The CHIL system is officially appraised in many evaluation campaigns, and the system is ranked among the best. One of Tekmo’s thesis comments was “The biggest problem in real environment acoustic event detection is overlapping – i.e. temporal intervals where the acoustic event of interest is overlapped with speech and/or another acoustic event. It was found that the overlapping segments account for more than 70% of errors produced by every submitted system” (Temko and Nadeu, 2007, p. 148).

2.3.3 Mixed Audio Classification

Classification is important in archive management, information mining from big data, and many other applications. No previous study has been general enough to propose a universal system that will maximise information retrieval for further information mining. Although the impact of overlapping classes might decrease the information retrieval system performance, there has been little research done on addressing this problem. Some examples of this are Thambi et al. (2014), Kos et al. (2013), Mohri et al. (2007), Seyerlehner et al. (2007), Temko1 et al. (2006), Chou and Gu (2001), El-Maleh

et al. (2000), Khonglah and Prasanna (2016), Khaldi et al. (2016). The detection of these classes with the presence of overlap conditions or real-world audio detection has seldom been studied. Zhang and Kuo (2001) presented an approach to segmentation and annotation of audio-visual recordings into three main categories, namely, silence, and with and without music components. Each of the last two categories were then further sub-classified. Four kinds of audio features were used, namely, the short-time energy function, the short-time average zero-crossing rate, the short-time fundamental frequency and the spectral peak tracks. The segmentation performance of the proposed method when dealing with music, speech with background music, and sound effects with background music was up to 94.5%, 86%, and 87.5% respectively.

For the case of real-world audio classification, universal open architecture has been proposed as a possible solution to the problem of non-exclusive classification (Duncan et al., 2014). The proposed system has the ability to detect speech, music, and event classes even where they overlap simultaneously and to label the input signal via a timestamp. Some overlapped classes are complex in nature. Hence, they cannot be straightforwardly detected by common features. Shokouhi et al. (2015) proposed an overlapped speech detection algorithm with a presence of noise condition to estimate the likelihood of overlapping speech. The spectral subtraction algorithm for speech enhancements is used to detect SNR from non-speech regions. The aim of the suggested algorithm is to estimate the word-count of the input audio file. Finally, syllable rates are estimated by dividing the total number of syllables by the segment length. Carrying the idea forward, MARSYAS software has been combined with the spectral subtraction algorithm, the output being plotted to the timestamp. This work was also an extension of the previous work (Mohammed et al., 2015).

For detecting the presence of music in noisy backgrounds, Lee and Ellis (2008) suggested a robust musical pitch detection algorithm, capable of dealing with highly variable environmental recordings such as the soundtracks of consumer video recordings, detecting music in ambient audio. The best music/speech discrimination result, with vocals present, of the proposed features was when it was combined with Rhythm and 4 Hz Modulation Energy (4HzE). Moreover, YouTube recording samples used to test the proposed feature achieved 91.4% accuracy when combined with a variance of spectral Flux and Rhythm. Tomonori et al. (2008) investigated four sets of features for background music detection. The first set is empirical features including 3-Hz modulation energy, the percentage of low-energy frames, the spectral centroid, the spectral roll-off point, the spectral flux, and the zero-crossing rate. The second set is Mel-frequency Cepstral Coefficients (MFCC). The third one is the linear frequency spectral powers feature, which is derived from the equally spaced band in the frequency domain. The short-time Fourier transform (STFT) is calculated for each band. Finally, the 80-dimensional feature vector is computed by averaging values within each frequency band. The last feature set is the spectral powers feature with the MFCC (SPMF). All features were extracted for every 50 ms-long frame. The experimental results show the first features set to provide the highest music detection accuracy with a high music-speech mixing ratio, whereas the third set reflects the highest results with a low music-speech ratio (-10 dB and -20dB). Boakye et al. (2008) explored about 40 features for detection of the overlapped speech to improve the accuracy of speaker dimerization. The system introduced classifies the speech recorded in the context of a meeting into three classes: non-speech, speech, and overlapped speech (more than one speaker speaking simultaneously). The author indicated that the best F1_Score performance was 47%.

2.3.4 Sound Source Separation

Speech/music separation algorithms are suggested as a possible solution to automatic speech/music discrimination and audio content analysis, and several techniques have been proposed in the literature. Even though there is no software available in the public domain - neither commercial nor free - able to accomplish speech/music separation with mono channel recording, giving adequate performance without prior knowledge about a given signal. By way of explanation, it is worth noting here that usually the best method to perform the separation with mono channel recording depends on the type of a priori knowledge which one has about the mixed audio signal, which of course is absent in the arbitrary sound recording (the case for this study). The domain is usually a time-frequency representation of the mixed signal. In the field of Blind Audio Source Separation methods, the assumption is that there is not any knowledge available about the input sources.

Techniques such as Independent Component Analysis (known as ICA) (Comon, 1994), Non-Negative Matrix Factorization (Lee and Seung, 2001), or statistical approaches like Hidden Markov Model (known as HMM), which work on the basis of prediction using previous data values, are generally employed (Comon and Jutten, 2010).

From a literature point of view, most primary BSS/ICA algorithms found in the literature are iterative. These range from the pioneering neural network approaches by Hérault et.al. (1985), Comon (1994), Macchi and Moreau (1997). In addition, one of the essential theoretical assumptions is that signal components are statistically independent of each other (Comon, 1994). The ICA methodology takes into account the structure of the covariance matrix of the dataset matrix that is under test. This is generally completed by means of a Singular Value Decomposition (SVD). This orthogonality only lets one discover the mixing matrix up to an orthogonal element (Comon, 2010, p. 155).

Regarding Non-Negative Matrix Factorization, which is a cluster of algorithms concerned with the decomposition of multivariate channels where the input matrix is factorised into more than one matrix, one should note that all the input and output matrices have no negative elements. This non-negativity property makes the decomposed matrices easier to inspect (Lee and Seung, 2001). Dissimilar to the BSS methods based on ICA, Non-Negative Matrix Factorization does not consider that the sources of the given multi-channel signal are independent. Non-negative matrix factorization techniques have also been applied to detect polyphonic sound events. An initial non-negative matrix factorization method for sound event detection was suggested by Heittola et al. (2011); the authors utilised NMF for the source separation stage and then applied HMM to improve the separated events through the prediction approach. An on-line NMF approach on the basis of the source separation has been covered in Joder et al. (2012). The authors developed a semi-supervised method for splitting the noise units in the observed frame using a sliding window. The prior knowledge about the speech bases is provided from a training dataset while the noise constituents are measured on-line in the recent past. Thus, the suggested work is performed based on the consideration that one source is known; the unknown sources are initialized randomly and updated with each new frame. Joder's approach in the preceding has been followed by sufficient work for speech/noise separation by Weninger et al. (2011) and Wilson et al. (2008), when the separation of both sources - speech and noise modules - through prior learning. In Smaragdis et al. (2017), a different approach has been taken for noise separation, where the authors considered that the frequency margins of the mixed sources are known via prior learning. Nevertheless, this estimation requires off-line handling, where the given sound data or noise type is known. By contrast, to the preceding and most conventional

methods that are used for time series forecasting, the SSA method is a statistical technique (non-parametric) and has no prior hypotheses or presumptions about the time series under consideration.

SSA is a tool for time series decomposition and has been deployed in a diverse range of analysis problems. It has been providing adequate results through its ability to decompose the time series based on the oscillation. There are a great many publications discussing aspects of SSA methodology and its applications. The range of these applications extends through trend detection and prediction, digital signal processing, image processing, health, geology, and psychology, some examples of which are Lu and Sanie (2015), Eftaxias et al.(2015), Zeng et al.(2014) and Ghil et al. (2002). It is used for tremor and climate prediction by Eftaxias et al. (2015) and Ghil et al. (2002) respectively; they reported the ability of the SSA method to extract the target components. In the frequency separation, Mert and Milnikov (2011), Harris and Yuan (2010) used SSA for separation of low-frequency components from the high frequencies. Depending on the same concepts, Zeng et al. (2014) deployed SSA for the elimination of environmental sound from heart sound signals through using eigenvalues to select the effective Principal Components (PCs). In image processing, an SSA spectrum was used to analyse a patient's movement disability's effect on their grasp, which is necessary to determine the type of therapy (e.g. Lee et al., 2013).

The most of the relevant work was in the classification and localising field. Mohammadi et al. deployed SSA for improving time–frequency domain sleep electroencephalography (EEG) classification. It is used as a pre-processing step to improve the analysis of EEG signal through separating the components which related to brain waves, sleep spindles and K-complexes (Mohammadi et al., 2016). Also, Enshaeifar (2016) applied

SSA for sleeping analysis. The authors introduce a new method for categorising sleeping into five levels through decomposition of EEG data by SSA. For improving the pulmonary auscultation, which is a widely used diagnostic method, and then separation and localising the heart sound, Sebastian and Rathnakara (2013), Ghaderi et al. (2011) deployed SSA on the respiratory data and introduce an adaptive method for selecting eigenvalues which correspond to heart sounds. Sanei et al., who is one of the SSA 'gurus', detected a murmur from heart sounds through changes in the statistical properties of the data, decomposed using SSA (Sanei et al., 2011b). Jarchi and Yang (2013) proposed a method for discriminating walking into three categories, namely walking downstairs, walking level, walking upstairs, and using ten healthy subjects. The sound was recorded using a triaxial sensor accelerometer positioned on the ear. SSA was used to decompose the time series and remove the noise trend from the signal. Moreover, SSA was used for eliminating the noise components at the feature extraction stage in hyperspectral imaging. The experiment shows that the distinction ability of the features has been much improved.

In this study, the use of the singular spectrum analysis algorithm has been investigated for localisation of the speech and music subspaces in the single mixed channel, which represents real-world audio and improves the classification results.

2.4 Conclusion

During the past 30 years, many more multimedia archives have become available. They have been used as essential references in many applications and disciplines. The localisation of audio classes, which is beneficial to information retrieval from audio content, was the main objective of researchers working in the audio content analysis area as mentioned in the above literature. A careful study of the literature reveals that:

- Overlapped soundtrack classification has not been thoroughly investigated. Also the overlapping problem represents the major challenge to achieving high performance in automatic audio classification.

- It should be noted that many classification regimes and algorithms such as ANNs, HMMs, SVMs, KNNs, and GMM with a diverse set of features were efficiently used for information retrieval of specific class detection with specific conditions. Hitherto, there have been limited studies that have addressed real-world audio problems that might contain speech, music, or a combination thereof; this has motivated the present study.

- In terms of feature space, literature reviews have indicated that were not many researchers who have used the automated audio features extraction methods such as the optimisation and the machine learning. In the area of audio classification or categorisation, the utilised audio Features (included low-level descriptors) were heuristically developed. All the related research is deeply influenced by psychoacoustic (sound perception and How are perceived by the human).

- A clear understanding of overlapped features is essential for detection of both the speech and the music occurrences. In addition, the success rates depend on the appropriate feature spaces and window lengths. This seems to suggest that zooming in signal processing, analysis, and the combined use of both short-term analytical frames and slightly longer-term segments are particularly beneficial.

- All the above techniques show promising results and have become popular. However, the random forest technique has outperformed other techniques in the audio classification area and shown the ability to learn effectively from large data samples with very few features, with promising classification results as shown in the

literature. Moreover, it provides stable outcomes and is robust against dataset noise.

- Singular Spectrum Analysis is used efficiently and widely for both noise removal and sound decomposition in diverse applications with significant improvement. In summary, SSA has not been considered before to solve the overlapped soundtracks problem in spite of the fact that SSA could be a potential solution due to its many interesting results and the ability of components separation.

Thus, the classification of overlapped audio soundtracks could be improved by applying a novel pre-processing step to the proposed classification system. The proposed algorithm depends on singular spectrum analysis and the Random Forests technique.

3 THE COMMON AUDIO FEATURES

3.1 Introduction

One of the most common facts in the field of audio classification, and yet also one of the outcomes from the preceding chapter, is the importance of the extracted feature and analysis stage. This is an essential process in pattern recognition and machine learning. Feature extraction implies the conversion of audio signals into a set of meaningful information called the feature space. In this study, the dataset is audio signals. They comprise an exceedingly large dataset, which is difficult to process directly. In such cases, it is necessary for the researcher to become familiar with audio features. Based on this knowledge, the feature sets can be heuristically selected.

In this chapter, the well-established audio features that have been used by many researchers, seen in a broad range of audio classification and recognition systems and used by Lartillot et al. (2008) in MIRtoolbox, are presented. MIRtoolbox is considered a popular toolbox, deployed and validated by many other researcher and cited in plentiful publications in the music categorisation and in the speech/music discrimination areas (e.g. Jiang, 2012 and Lika, 2014). Some examples of this are McKay (2010), Michalevsky et al (2014), Knox et al. (2011), Collins et al. (2014) and Aubé et al (2014).

Furthermore, this chapter demonstrates the method the extraction and analysis of the deployed feature in the presence of the overlapping between audio classes. The features analysis stage comprises three steps. Firstly, mixing of pure samples (speech and music) to generate mixed soundtracks. Then, all features are calculated for pure and new mixed samples. Finally, comparisons between the extracted features for both cases (before and after mixing) are conducted through plotting them in a sequential manner. The calculation method of the fully developed features will in addition be described. All the adopted

features are deployed in the classification of the deployed dataset samples into speech or music regardless of whether these samples were pure or overlapped. It is worthwhile to note once again that the main purpose of this study is not to cover all audio features that are used, as described in the literature, but rather, the goal is to provide an adequate method to reduce classification obstacles where overlapping takes place between the audio classes.

3.2 Feature Extraction

Features are usually extracted from overlapping frames (each comprising a number of samples) that denote periodically or quasi-stationary characteristics, instead of estimating them over the whole signal *s(n)* or the spectrum calculated from the whole signal which greatly changes over time. Section 2.2.1 includes a list of some researchers who tested different frame sizes in their research; they show the effects of the frame size on the results. In this study, the long audio sample is broken into shorter frames of 50 ms long and 25 ms overlapping using a moving window technique.

The notations below will be utilised in the time domain:

- *s(n)* is the sampled audio signal.
- *n* represents the time index of the audio signal samples.
- N_t denotes the total length of the given audio signal in number of samples.
- *Fs* is the sampling rate of the given audio signal.

The following notations will be used for the time frames:

- *f(n) is* the sampled audio frame.
- *n* is the time index of the audio frame sample.
- *NF* is the total number of frames in the audio signal *s(n)*

- *L* denotes the length of the frame (total number of samples in each frame).
- *i*,(1≤ i ≤ *NF*) is the index of the frame position in the audio signal *s(n)*
- *hop* number of time samples between two successive frames (the size of window shifting measured in samples).

The ***s(n)*** is divided into a number of frames *f* equal to *NF*, where all frames have the same length *L*, $0 < L \leq N_t$, *L*= [duration time in seconds x *Fs*]. Figure 3-1 depicts the procedure followed and the corresponding notations.

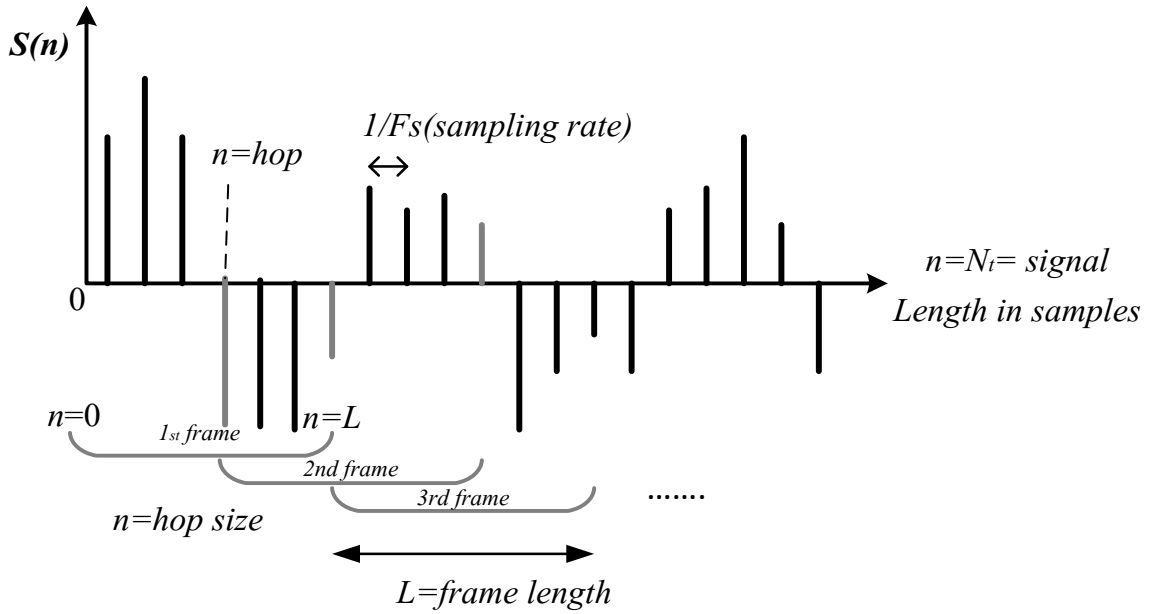


Figure 3-1 notations for audio signal framing

The Equation 3-1 can be used to calculate the number of frames (*NF*)

$$NF = \left\lfloor \left(N_t - L \right) / hop \right\rfloor + 1 \qquad 3\text{-}1$$

The hop size *hop* reflects the overlapping between the adjacent frames (calculated in samples) as illustrated earlier. Processing such as this produces a two-dimensional matrix *L* x *NF*, where each column corresponds to a separate frame sequence. The next step is the feature extraction: a set of time domain and frequency domain features is extracted from each frame (column), whereof the output is called the feature vectors or

feature space. The dimensionality of the generated feature vectors depends on the number of extracted features. In this study, 34 features are computed using MIRtoolbox, mentioned earlier, for each frame. These extracted features are presented in the following two sections. As mentioned above, audio features that have been used by many researchers and seen in a broad range of audio classification and recognition systems are extracted and deployed in this study.

3.3 Time Domain Features

Time domain features represent how waveforms change over time. In general, these features are calculated directly from the samples of the given sound signal. Such a property is considered the simplest method for sound analysis, and it is significant for the enhancement of results by combination with features that are more sophisticated. The following sections describe the features employed.

3.3.1 Zero Crossing Rate (ZCR)

Zero Crossing Rate represents the number of times that an audio waveform crosses the zero axis in a given time interval divided by the frame's length to remove the dependency on the duration. In other words, the ZCR refers to the number of signal sign changes per unit time. It is a robust feature used to discriminate between music and speech or between speech and non-speech samples (Chen, 1988). It is a straightforward computational method, employed in many studies related to such objectives as detection of speech and music, e.g. (Al-Maathidi and Li, 2012, Shete and Patil, 2014, Bachu et al., 2008, Chou and Gu, 2001), and automatic speaker recognition (Chen, 1988).

$$ZCR(i) = \frac{1}{2}\left(\sum_{n=1}^{L} Sign\left(f_i(n) \right) - Sign\left(f_i(n-1) \right) \right) \frac{Fs}{L} \qquad 3\text{-}2$$

where i is the frame's position in sequence of the audio signal $s(n)$.

$$Sign(x) = \begin{cases} 1 & if \quad x > 0 \\ 0 & if \quad x = 0 \\ -1 & if \quad x < 0 \end{cases} \qquad 3\text{-}3$$

The extracted feature will be presented as follows: at the beginning of the process, the features were extracted from three classes, namely Music (M), Speech(S), and Speech over Music (SM). Two figures are used to explain each feature and its distribution. The first figure comprises the following: 150 randomly selected feature samples from a dataset of calculated features were used for each of the aforementioned classes. The chosen samples were retained in a sequential fashion to allow further examination of their homogeneity. The first 50 samples are music, followed by 50 samples pure speech and then followed by the last 50 samples which are a mixture of speech and music. The Y-axis denotes the amplitude of the calculated feature, which is extracted from frames of 50 ms length, while the X-axis refers to the frame index. The following procedure is applied for a comparison between the presence of speech and music. An alternative statistic, which is the square of the standard deviation (σ^2) normalised to the square of mean (μ) value ($\frac{\sigma^2}{\mu^2}$) is computed for 1 second mid-term (for each 20 successive frames of the pre-calculated feature of speech and music) with a window which is moved by only one sample each time. This statistical feature was applied on a Root Mean Square (RMS) basis for the first time by Panagiotakis and Tziritas (2005) as a speech/music discriminator feature, where the authors refer to it as the "volume of invariant". Later on, the same features and representation has been used by (Giannakopoulos, 2014) .The second figure will present a histogram of the above statistical feature and in a similar presentation method.

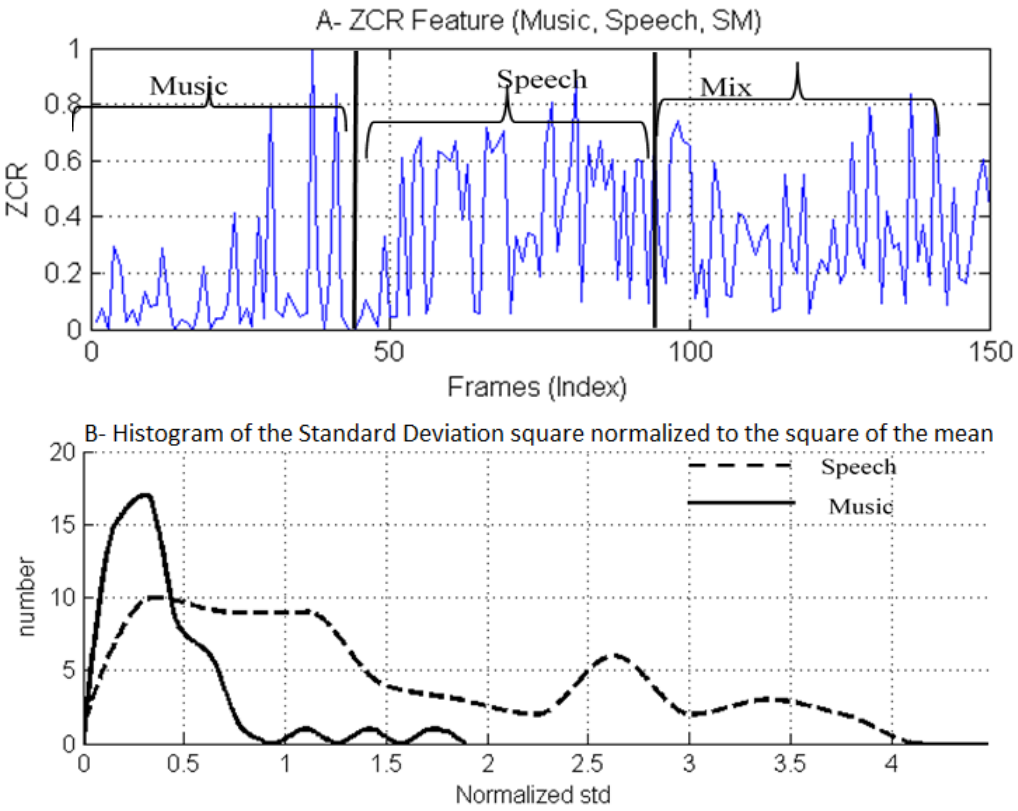


Figure 3-2 ZCR Feature. A- Calculated Feature in sequential for audio classes (M, S, SM) B- Histograms of the normalised standard deviation of the ZCR for music and speech classes.

Speech frames mostly have higher ZCR values than music because there are silence gaps amidst the continuous utterance and this manifests itself in abrupt changes from the point of view of the analysis. This is shown in Figure 3-2 (A), where pure speech samples generally have higher ZCR than other classes and thus appear to show wide-ranging changes between low and high power due to these silence gaps. For the same reason, the histogram Figure 3-2 (B), which presents the histogram of the statistical variables of ZCR of speech and music frames, indicates that ZCR values, which correspond to speech samples, vary over a greater range of values along both axes. This finding is consistent with many other studies, some examples of which are Shete and Patil, (2014), Bachu et al. (2008) and Panagiotakis et al. (2005). This makes ZCR a useful feature to discriminate between pure speech and music. Nevertheless, it is not

adequate in the presence of overlapping between audio classes. This is a common observation, and it has a physical meaning since speech samples have fewer silence gaps when they overlap with other classes, i.e., the silence intervals between utterances will contain music rather than silence in the case of overlapping with music.

3.3.2 Root Mean Square (RMS)

The calculated RMS of the audio signal is usually expressed in decibels. It was defined for the first time by Kenny and Keeping (1962). It is a commonly used feature. Tzanetakis et al. mention that frames with silence have lower RMS than those without silence (Tzanetakis and Cook, 2002). This feature is defined according to Equation 3-4 by Kenny (1962).

$$\mathrm{RMS}(i)=\sqrt{\frac{1}{L}\sum_{n=1}^{L} f_i(n)^2} \qquad 3\text{-}4$$

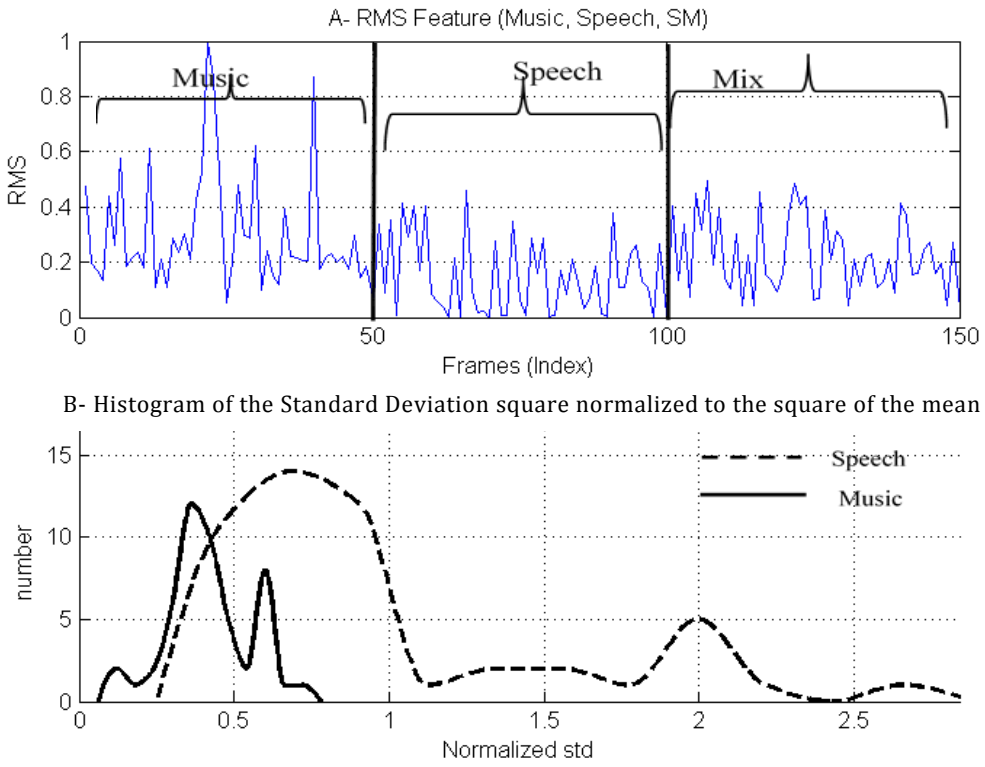


Figure 3-3 RMS Feature value, A- Feature Amplitude in sequence for audio classes (M, S, SM), B- histogram of RMS values for speech and music classes

From the data in Figure 3-3 (A), it can be seen that music and mixed classes generally have higher values than those which have speech context. The histogram in Figure 3-3 (B) indicates that the statistical variable of speech is slightly higher than that of music and the latter has a wider distribution range and lower level of invariant.

3.3.3 Entropy

The entropy feature is a measure of information averaged over the frame, and reflects the randomness of the signal (entropy increasing with randomness, which reflects a higher amount of information and vice versa) (Shannon, 1948). The default mathematical definition is shown in Equation 3-5 (Shannon, 1948):

$$H = -\frac{1}{\log_2(L)}\left[\sum_{i=1}^{L} Pr(x_i)\log_2\left(Pr(x_i)\right)\right] \qquad 3\text{-}5$$

The value of *Pr(xi)* is the probability that *f(n) = xi* and *xi* represents the sample space of the given frame, see Section 7.1.1 for more information about the probability calculation method.

Consistent with the conclusions of Misra, who illustrated that voiced sound would have lower entropy than noisy or non-speech, which corresponds to a flatter spectrum (Misra et al., 2004), Figure 3-4 (B) indicates that pure classes with silence intervals such as speech frames have a higher invariant level of entropy. However, the overlapped speech with music, or even the pure music, could reflect a lower invariant in the entropy.

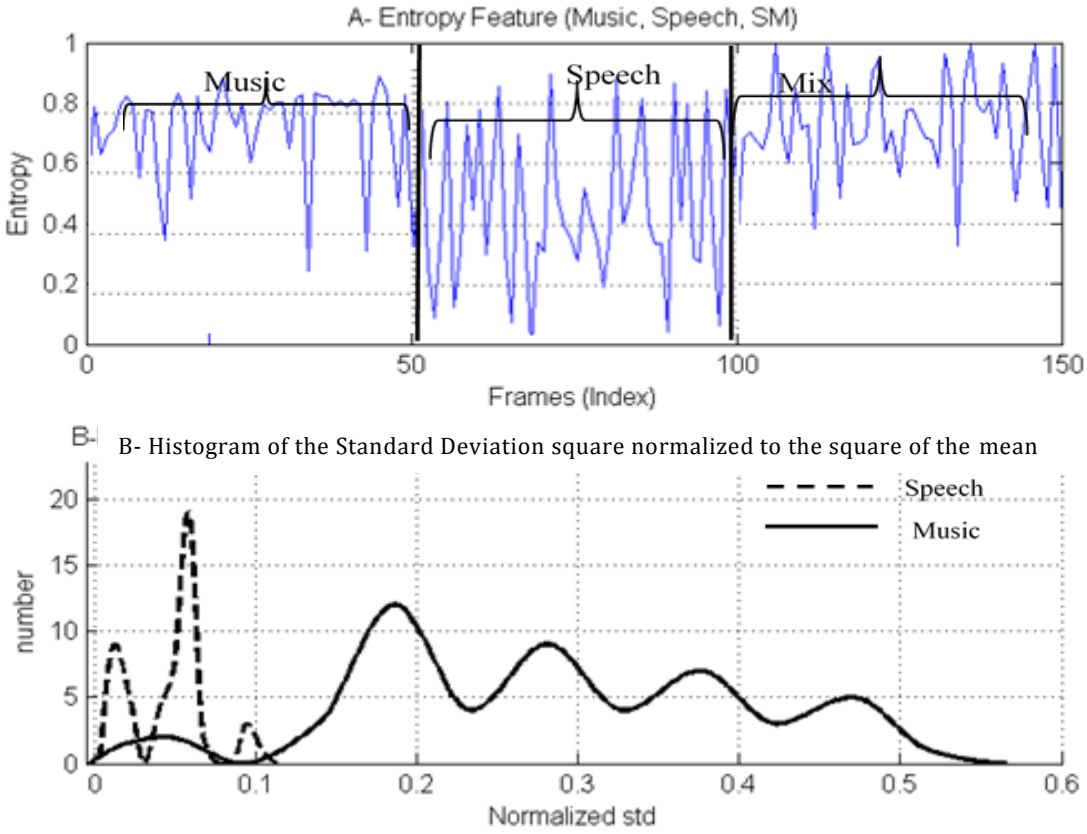


Figure 3-4 Entropy Feature, A- Feature Amplitude for (M, S, SM), B- distribution histogram of with speech and with music segments for entropy values

3.4 Frequency Domain Features

Frequency domain features are also known as spectral domain features and represent the spectral distribution of the signal. Frequency domain features can be calculated by Discrete Fourier Transform, where the output is decomposed into two parts. The first, the magnitude, represents a measure of the intensity and corresponds to the participation of the frequency in the input signal. The other part represents the phase of the signal. Another transform method is the Discrete Cosine Transform, which expresses a finite duration signal as a weighted sum of cosines of real numbers. Fast Fourier Transform, which exploits computational redundancy in the equation that defines the DFT. It is an efficient algorithm for finding the results of the DFT, (Giannakopoulos, 2014, p.33-36).

The notations below have been used in the demonstration of spectral domain features:

- *k* represents the bin index of the frequency.
- *S(k)* denotes the audio spectrum calculated for the audio frame using FFT.
- *P(k)* reflects the squared magnitude audio spectrum estimated for the *S(k)* of the audio frame.
- N_{FT} is the size of the spectrum.

The framing pre-processes explained in the time domain features section correlate to a multiplication of a given audio signal *s(n)* by a rectangular window function for the frame $f(_i)$. Meanwhile, the multiplication in the frequency domain corresponds to the convolution (*) operation. Hence, to avoid the distortion of the spectrum, the applied window function should be represented by a very narrow main maximum with side values close to zero maxima such as the Hanning window (see Section 2.3.1). Consequently, the frame in the i^{th} position is calculated according to Equation 3-6.

$$f(i) = 0.5\left[1-\cos(2\pi\frac{\mathbf{n}}{L-1})\right]\left(s(j+\mathbf{n})\right) \Rightarrow \mathbf{n} = 0...L-1 \qquad 3\text{-}6$$

where the first part of the equation symbolises the Hanning window and *j* is the sample index of the frame in the i^{th}.

3.4.1 Pitch

Pitch refers to the fundamental frequency (F_0) which is considered the primary key to detecting harmonics. Therefore, it is necessary for the segmentation process - as well as analysis and synthesis of speech and music - Normally only voiced speech and harmonic music have well-defined pitch.

3.4.2 Brightness

The brightness describes the spectral distribution of the frequency over the sound segment, through measuring the proportion of frequencies above the cut-off frequency, which is equal to 3000 Hz (Juslin, 2000). Lartillot et al (2008) reported that the default value of cut-off frequency is 1500 Hz, whilst Laukka et al. (2005) have suggested another value of the cut-off frequency which is 1000 Hz. In this study, 1500 Hz has been used as a cut-off frequency as a mid-value between the high and low frequencies that represent music and speech respectively. Consequently, the samples with music appearing are represented by higher brightness and those that are without music represented by a lower brightness value. In other words, brightness detects whether the signal is represented by high or low frequencies (Mitrovic et al., 2010). Figure 3-5 demonstrates the calculation method of the brightness value.

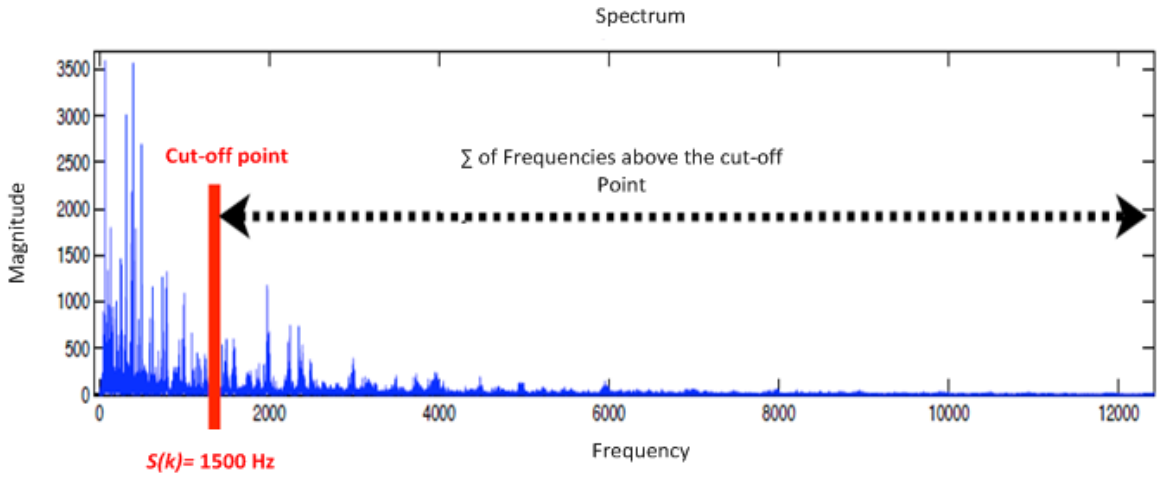


Figure 3-5 Brightness Calculation Procedure

The default mathematical definition is shown in Equation 3-7 (Juslin, 2000):

$$Brightness = \frac{\sum_{k=cutoff-point}^{N_{FT}/2} S(k)}{\sum_{k=1}^{N_{FT}/2} S(k)} \qquad 3\text{-}7$$

Brightness has been applied in many audio content analysis studies through detection of the frequency distribution (Liu and Wan, 2001, Wyse and Smoliar, 1995, Scheirer and Slaney, 1997). The brightness of the classes depicted in Figure 3-6 indicates that the brightness level of music is higher and less extensive compared with that of speech, and for this reason the histogram of the speech group ranges far across the domain of values. To reduce the sophistication of the processing, this feature has been deployed with other combinations of features to categorically identify music.

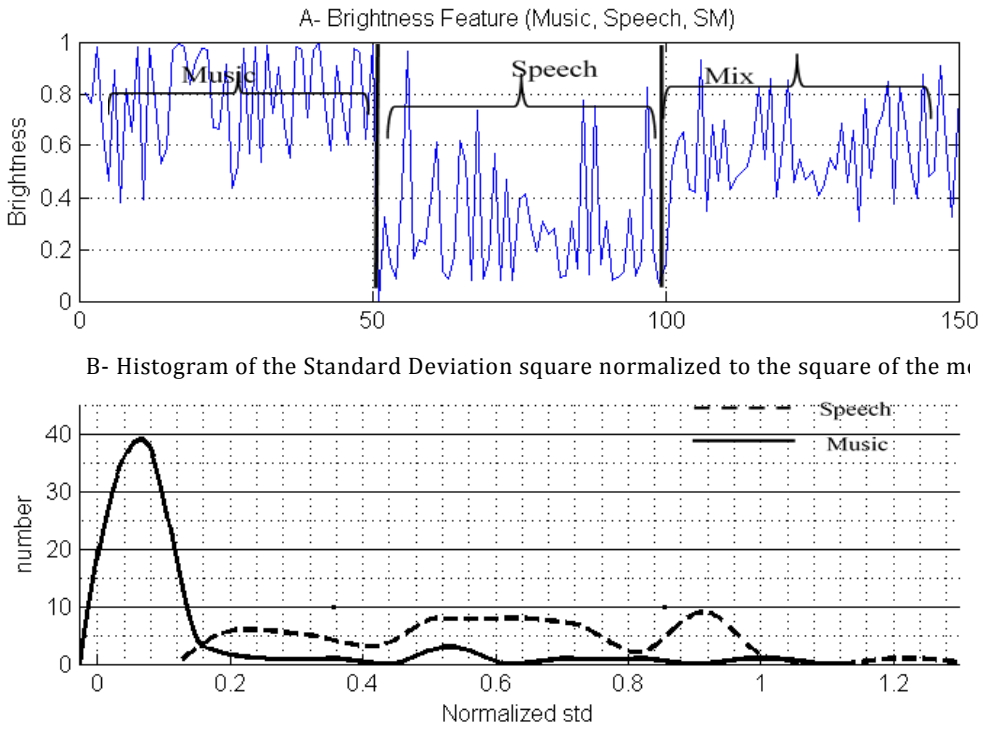


Figure 3-6 Brightness Feature, A- Brightness Amplitude (M, S, SM), B- distribution histogram of with speech and with music segments for Brightness values

3.4.3 Roughness

The roughness feature was introduced for the first time by Plomp and Levelt (1965) as an approximation of the sensory dissonance. It is calculated by computing the peaks of the frequency, and then calculating the mean of all the dissonances between all potential pairs of peaks (Sethares, 1998).

It is apparent from Figure 3-7 (A) that the roughness of pure speech samples is less pronounced than that of music and speech over music samples because it includes spectrally coherent fluctuations or higher abrupt changes ratios between silent and voiced gaps. For the same reason, the music samples reflect slightly more invariant range than other classes as apparent in Figure 3-7 (B). This finding is consistent with the findings of others (e.g. Fleischer, 1976).

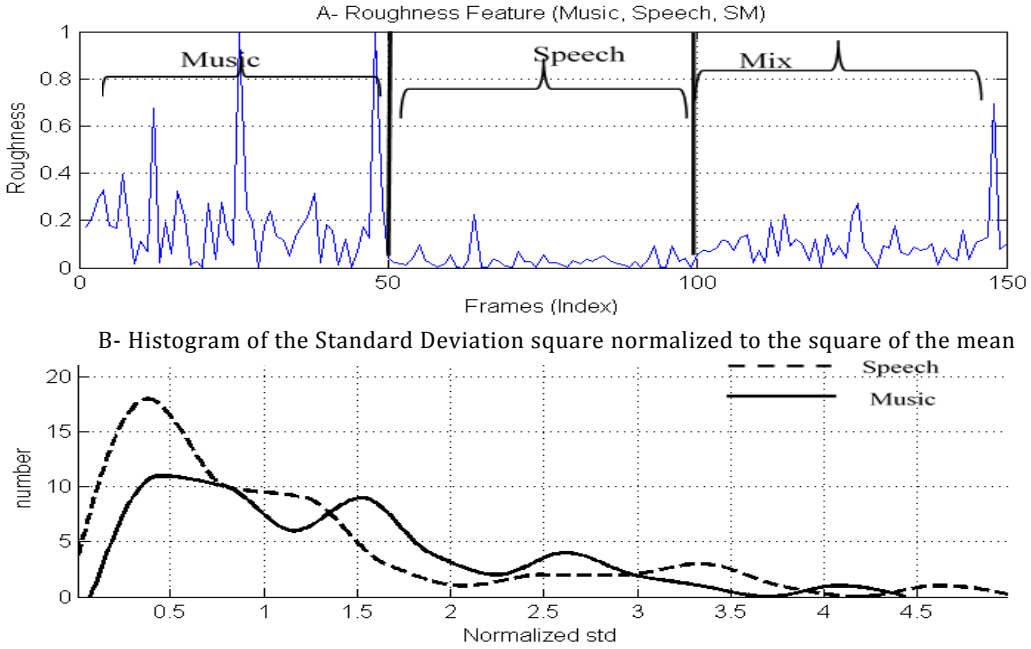


Figure 3-7 Roughness Feature, A- Roughness Amplitude (M, S, SM), B- distribution histogram for speech and music class segments

3.4.4 Irregularity

The irregularity spectral feature represents the degree of dissimilarity between consecutive peaks of the frame spectrum. It was proposed for the first time by Jensen (1999). The total irregularity is estimated as the quotient of the sums of the square of the variance between the consecutive spectrum bins and the square of the total spectrum as apparent in Equation 3-8 (Sethares, 1998):

$$Irregularity = \frac{\sum_{k=1}^{N_{FT}/2} \left(S(k) - S(k+1)\right)^2}{\sum_{k=1}^{N_{FT}/2} \left(S(k)\right)^2} \quad 3\text{-}8$$

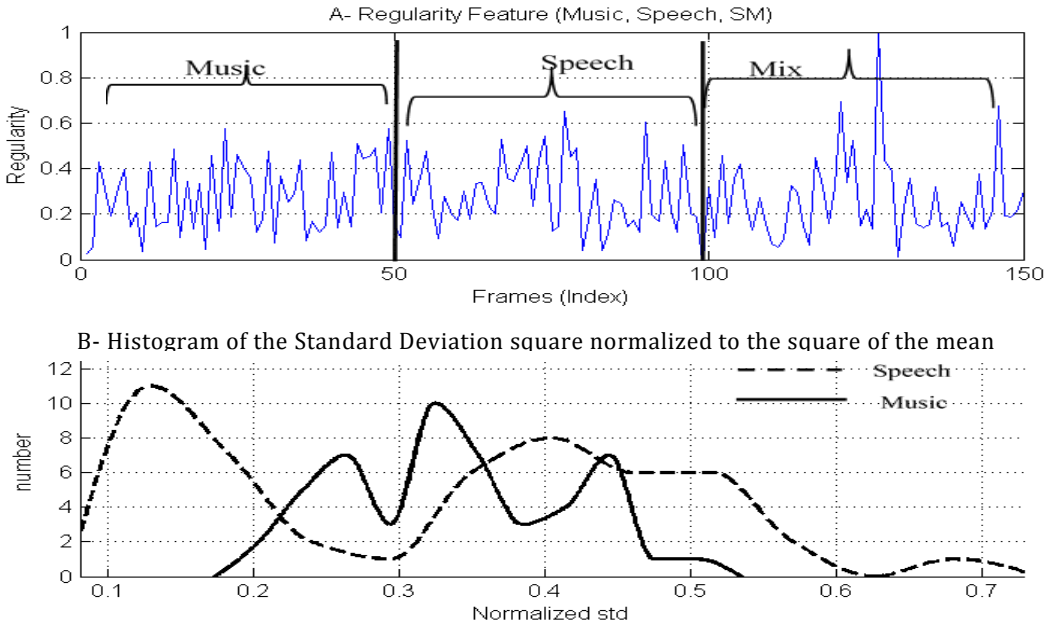


Figure 3-8 Spectral Irregularity, A- Feature Amplitude Sequence (M, S, SM), B- Histogram of std for speech and music segments.

In Figure 3-8, the histograms of the normalised std value of the sequence of values of the irregularity feature for audio segments extracted from speech and music classes is demonstrated. The example specifies that the speech samples reflect a slightly wider spread than the music class.

3.4.5 Spectral Roll-off Frequency

Spectral Roll-off point Frequency denotes the frequency below which 85% of the spectrum magnitude can be found. Most of the signal power is concentrated within a certain range of frequencies. It increases with increasing bandwidth and is used for music information retrieval as well as to discriminate between music and speech (Tomonori et al., 2008). It was also used by Scheirer and Slaney (1997) to distinguish voiced from

unvoiced sound. Palo et al. (2015) developed a new feature for emotional speech recognition based on spectral roll-off by finding the sub-band frequency. The spectral roll-off feature can be computed as specified in Equation 3-9 (Kim et al., 2005):

$$\sum_{k=0}^{Kroll} |S(k)| = 0.85 \sum_{k=0}^{N_{FT}/2} |S(k)| \qquad 3\text{-}9$$

where *Kroll* is the frequency bin corresponding to the estimated roll-off frequency. Along with the others, the data plotted in Figure 3-9 (A) shows music or mixed sound may show marginally greater values of this parameter because samples of this kind hold more information, representing more power. By contrast, the class with speech present demonstrates slightly higher and more extensive values of the variant in the frequency domain as demonstrated in Figure 3-9 (B).

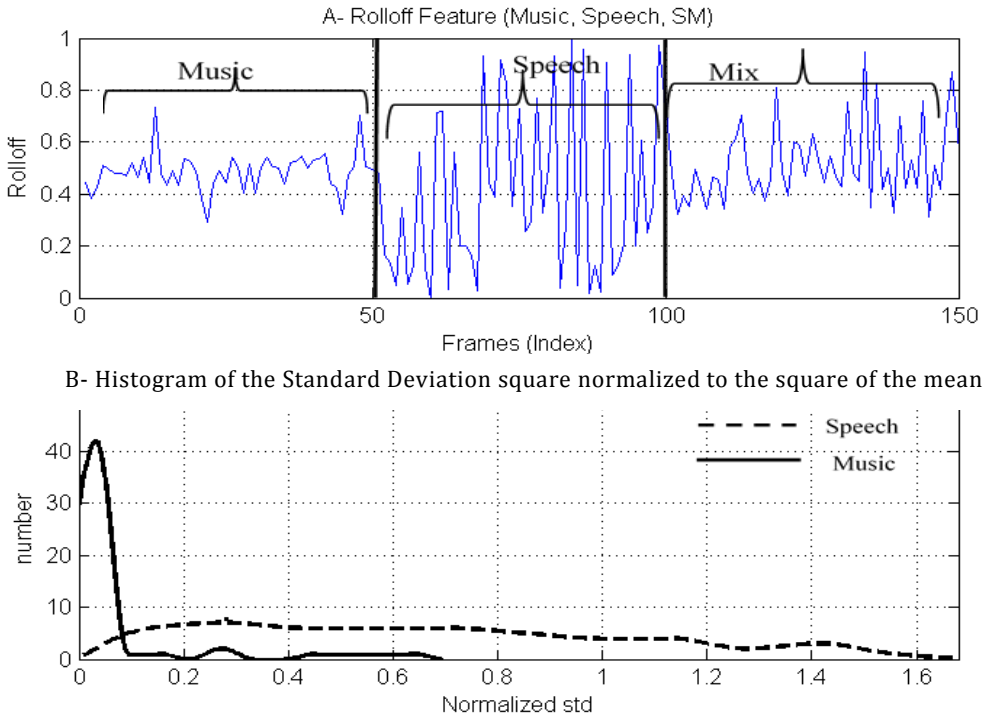


Figure 3-9 Spectral Roll-off, A- Feature Amplitude for Sequence (M, S, SM), B- distribution of with speech and with music segments for Roll-off values

3.4.6 Spectral Centroid (SC)

Spectral Centroid is defined as the centre of the gravity of the spectrum magnitude; it

corresponds to determining the spectrum point where the most energy is concentrated within the audio frame (Kim et al., 2005). Therefore, SC reflects the spectral shape of the frame. Mitrovic et al. (2012) refer to SC as a rough calculation of the brightness. The value of SC for the i^{th} frame can be calculated as defined in Equation 3-10 (Kim et al., 2005):

$$SC(i) = \frac{\sum_{k=1}^{N_{FT}/2} \left(kP(k)\right)}{\sum_{k=1}^{N_{FT}/2} \left(P(k)\right)} \qquad 3\text{-}10$$

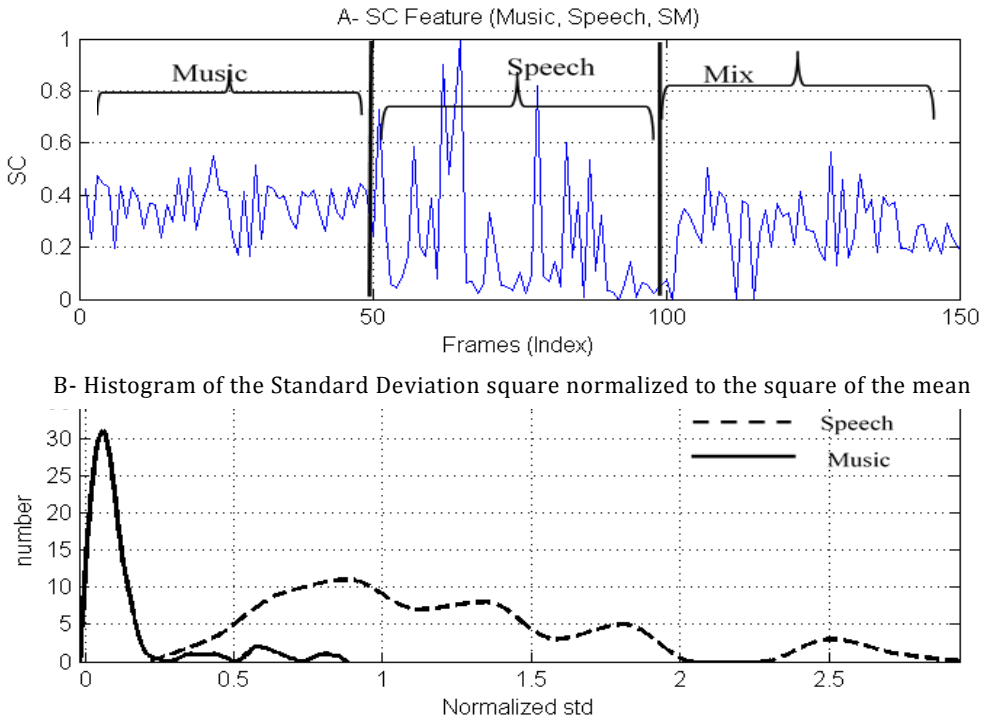


Figure 3-10 Spectral Centroid Feature, A- Feature Amplitude Sequence (M, SM, S), B- Histogram of with speech and with music classes.

In Figure 3-10-B, the histograms of the spectral centroid value for audio segments from two datasets -speech and music groups - are presented. It can be seen that the 'with music' class typically includes lower higher invariant and less extension range for the calculated statistic, while the values corresponding to the with speech group are wider.

3.4.7 Spectral Spread (SS)

Spectral spread is defined by MPEG-7 as the second central moment of the spectrum, see Section 2.5.3 in Kim et al. (2005). The SS for a given frame is calculated by measuring the deviation of the spectrum magnitude from the SC of the same frame according to Equation 3-11 (Kim et al., 2005):

$$SS(i) = \sqrt{\frac{\sum_{k=1}^{N_{FT}/2} \left(k - SC_i\right)^2 P(k)}{\sum_{k=1}^{N_{FT}/2} \left(P(k)\right)}} \qquad 3\text{-}11$$

Indeed, SS reflects the shape of frequency around the corresponding centroid. Giannakopoulos (2014) mentioned that a highly centred spectrum of a given frame around the SC value is represented by low values of the SS. An interesting observation from Figure 3-11 (A) is that the value of this feature is tighter and of lower frequency for music and mix segments than for pure speech. Consequently, in Figure 3-11 (B) the histograms of the standard deviation of SS over the mid-term frames of music and speech segments are presented; it shows that the std value of this feature is higher and wider for speech than for mixed and music segments. The reason for this is that the SS value for speech has lower invariant than music samples.

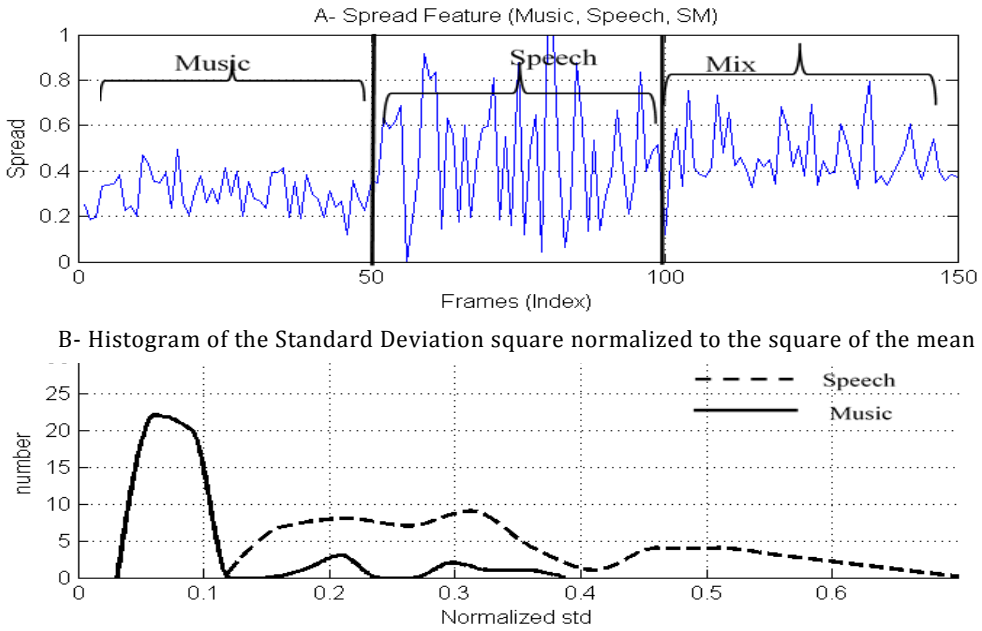


Figure 3-11 Spectral Spread SS Feature, A- Feature Amplitude Sequence (M, SM, S), B- Histogram for with speech and with music classes.

3.4.8 Spectral Skewness

Spectral Skewness represents the third order central moment, and as a consequence is measured as shown in Equation 3-12, (Eyben, 2016):

$$S_{Skewnes} = \frac{\sum_{k=1}^{N_{FT}/2} \left(k - SC_i\right)^3 P(k)}{SS^3 \sum_{k=1}^{N_{FT}/2} \left(P(k)\right)} \qquad 3\text{-}12$$

In Figure 3-12 (B), the histograms of the spectral skewness value for audio segments from two datasets - speech and music groups - are presented. It can be seen that the 'with music' group typically includes lower amplitude and less extension for the calculated statistic, while the values corresponding to the speech segments are higher as a consequence of calculated Spectral Skewness based on the spectral centroid feature.

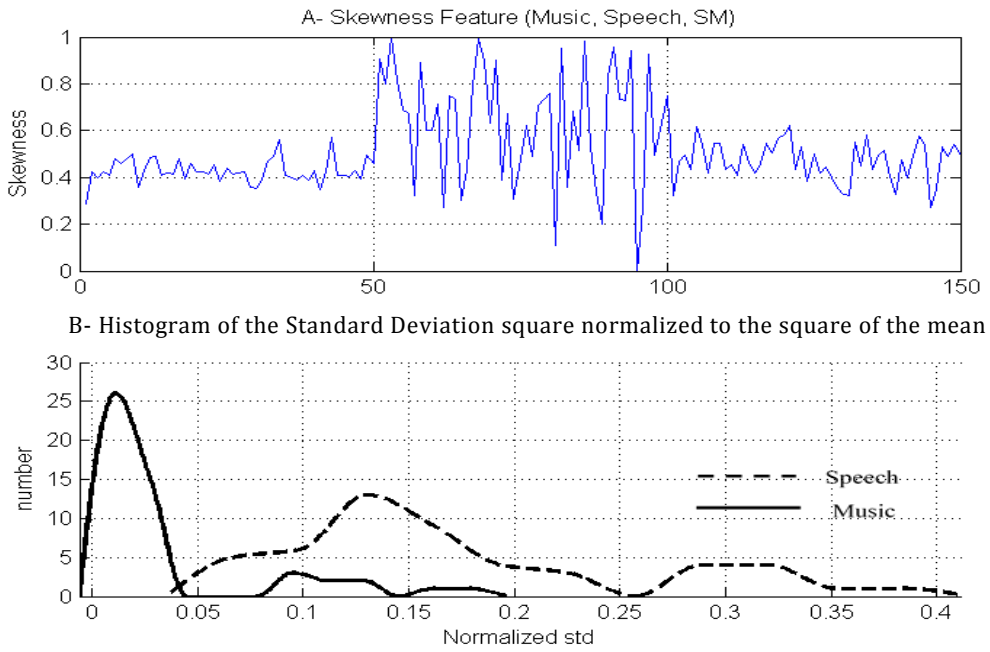


Figure 3-12 Spectral Skewness Feature, A-Feature Amplitude Sequence (M, SM, S), B- Histogram of standard deviation for speech and music segments.

3.4.9 Mel Frequency Cepstrum Coefficients (MFCC)

MFCC is one of the most common audio features, used in many audio applications (speech recognition, speaker recognition, sound classification, and MIR). Davis and Mermelstein (1980) designed MFCC parameters primarily for speech recognition, and these parameters have been shown successful in capturing significant acoustic information. It is shown to be effective in a diverse range of classification systems, for example Bae et al., (2008) and Weng et al., (2010).

MFCC is computed using a number of steps (Kim et al., 2005, p. 52-55) (Bae et al., 2008), which can be summarised as follows:

- The Fast Fourier Transform (FFT) is calculated to transform the time-domain data to the frequency domain for each frame.
- The absolute value is taken to obtain the magnitude spectrum.
- Mel frequency scale, which is a unit of pitch used to convert a frequency *f* in hertz

into its equivalent value in *mel*, is measured through Equation 3-13 (Kim et al., 2005):

$$Mel = 1127.0 \log\left(1 + \frac{f(HZ)}{700}\right) \quad 3\text{-}13$$

- The reduced spectrum is measured by handling the spectrum via a triangle Mel-filter bank.
- The spectrum log energy of the reduced spectrum within the pass band of each filter is calculated.
- Finally, MFCC is obtained by calculating the discrete cosine transform (DCT) of the reduced log energy spectrum.

$$C_{co} = \sum_{j=1}^{N_f} \log(E_j) \cos\left[\left(j - \frac{1}{2}\right)\frac{i\pi}{N_f}\right] \quad 3\text{-}14$$

where E_j refers to the spectral energy calculated in the band of the j^{th} Mel filter and N_f represents the total number of Mel triangular filters in the bank (frequently N_f equal to 24) while, $1 \leq co \leq N_c$, N_c is the total number of applied cepstral coefficients C_{co} which are extracted from each window frame. The default value for N_c is 12, (Eyben, 2016, Kim et al., 2005).

3.4.10 Spectral Entropy

As described previously, Spectral Entropy was calculated for the first time by Misra et al. (2004), who calculated the entropy of audio spectra for the purpose of discriminating clean speech versus noisy speech; it is proposed as a feature for robust ASR. Misra had shown that clean speech has a lower level of spectral entropy than noisy speech because the mean number of abrupt changes is higher in a noisy environment. This feature is

generally estimated through calculating the entropy of an audio spectrum. It is also applied to ASR (Misra et al., 2004). It can be calculated by means of the following steps (Misra et al., 2004):

- Compute the power spectrum of the frame $P_i(k)$.
- In order to convert the spectrum into a probability mass function each frequency bin is normalised by the total spectrum power. Hence, the sum of area under the spectrum will be equal to 1 (Misra et al., 2004):

$$\bar{P}_i(k) = \frac{P_i(k)}{\sum_{k=1}^{N_{FT}/2} P_i(k)}, k = 1 \ldots \frac{N_{FT}}{2} \qquad 3\text{-}15$$

- Finally, the spectral entropy is computed according to Equation 3-16

$$H_i = -\sum_{k=1}^{N_{FT}/2} \bar{P}_i(k).\log_2(\bar{P}_i(k)) \qquad 3\text{-}16$$

The data trend in Figure 3-13 is consistent with the findings of past studies by Misra, who highlighted that speech samples have a lower value of spectral entropy. However, this hypothesis is rejected where speech samples overlap with music, as shown in the second and third groups in the Figure 3-13. The logical interpretation of this discrepancy is that the randomness of speech frequency as calculated stems from the music content rather than the speech content.

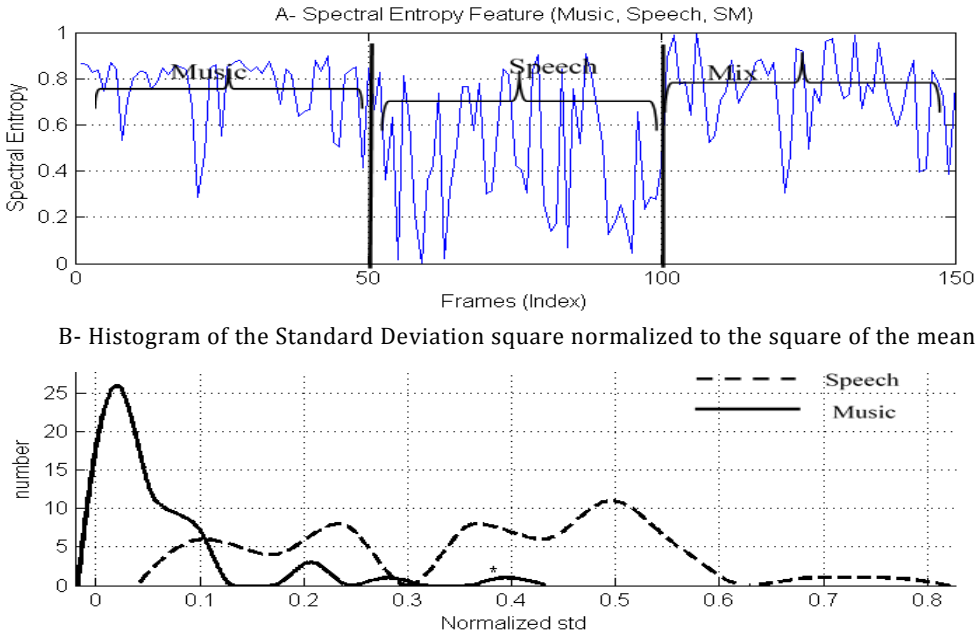


Figure 3-13 Spectral Entropy Feature, A-Feature Amplitude Sequence (M, SM, S), B-Histogram of standard deviation for speech and music segments.

3.5 Summary

In this chapter, the well-established audio features that in employed in this study are clarified. In addition, this chapter demonstrates the extraction and analysis method of these features in the presence of the overlapping between audio classes with reference to speech and music. All the explained features are used in the classification of the dataset samples into speech or music regardless of whether these samples were pure or overlapped. The next Chapter and Chapter 7 will illustrate how the explained features will organized and applied in the classification process.

4 RANDOM FORESTS

4.1 Introduction

The purpose of this chapter is to review the widespread and successful decision tree techniques of pattern recognition that are to be employed in this study. This chapter will be a supportive presentation of knowledge to provide a basic understanding of machine techniques that are used in the system proposed in the subsequent chapters. In the present chapter, the primary concern is with illustrating the Decision Tree technique in general and more specifically the Random Forests technique which has been used to develop the non-exclusive classification that is highlighted in Chapter 6 as a solution for overlapped content detection.

4.2 Decision Tree Overview

During the course of the last three decades, the DT has become one of the most frequently used machine learning techniques for supervised classification. Figure 4-1 depicts a simple DT with internal decision nodes and leaf nodes that represent speech/music categorization, the conclusion. In this example, a theoretical audio frame classification has been made; the DT at the beginning checks the RMS feature of the given frame. If the RMS power feature is less than a specific threshold then it will be considered unvoiced, otherwise it will be classified as a voice sample. Furthermore, the right node checks the ZCR value. If the ZCR is less than the specific threshold then it will be classified as music, otherwise it will be classified as speech.

This approach is known as top-down induction of DT. The first innovation of the DT algorithm was represented by Hunt's concept learning system framework in 1966

(Quinlan, 1986), which built a DT to minimise the classifying cost through measurement of two components: detecting the cost value of the object; and the cost of the misclassification. The concept learning system chooses an action to minimise cost in this limited space, then moves down a level in the DT (Quinlan, 1986). Many other algorithms have since been developed from the concept learning system.

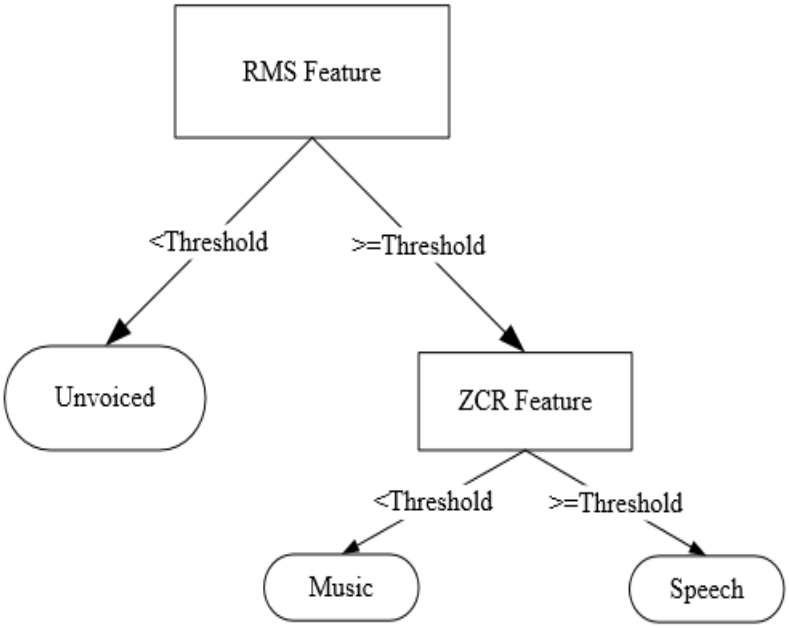


Figure 4-1 simple decision tree for audio file classification

The DT algorithm, which will be represented in the following sections, will use the training vectors (a set of data) and build a DT similar to the one in the Figure 4-1. DTs have been used in expert systems; the results were adequate and, indeed, nearly similar to those from a researcher expert. Harrington has listed the advantages of the DT as an inexpensive calculation in terms of computational power and not being too complicated for the user to understand the technical results. Giannakopoulos summarised the advantages of the DT in the following points (Giannakopoulos, 2014):

- The structure of the DT can be written as a set of rules without any impediments.
- Easy to handle and implements simple features.

- It is considered a non-parametric approach through handling the features without needing to make distribution assumptions.
- However, at the same time some authors referred to the sensitivity of the DT approach to noise that could make it unstable, depending on the dataset (Maimon and Rokach, 2008, Giannakopoulos, 2014).

In general, the DT can be described as follows (Maimon and Rokach, 2008, Giannakopoulos, 2014):

- The root has no input edge (output only) and generally, it holds the most important feature of the object (the features reflect the salient information of an object).
- The branches (internal nodes, non-leaf), with only one incoming edge and two or more outputs are the input attributes that lead to these classes.
- In a binary DT, each node has only two outputs.
- Terminal node or leaf represents a class label and with only one input edge.
- A new test sample is classified by passing it through the tree model that was built in the training phase, from the root to the leaf (final class); the outcome of the answer at the non-leaf nodes depends on navigation.
- Each root or internal node is represented by a single feature with its corresponding value or array of values which that feature might take.
- Each terminal node or leaf is denoted by a class label.

Let us assume that training vectors and target vectors (labels) are represented by **x** and **y** respectively, $v = \{\mathbf{x}_i, \mathbf{y}_i\} \Rightarrow i = 1, \ldots m$ where m is the number of the training samples. Based on the structure of the decision tree, the i^{th} test sample rests at its last destination space

d(t) at the terminal node t as illustrated in Figure 4-2, where the subspace d(t) is linked with nodes t for the same decision tree which was depicted previously in Figure 4-1.

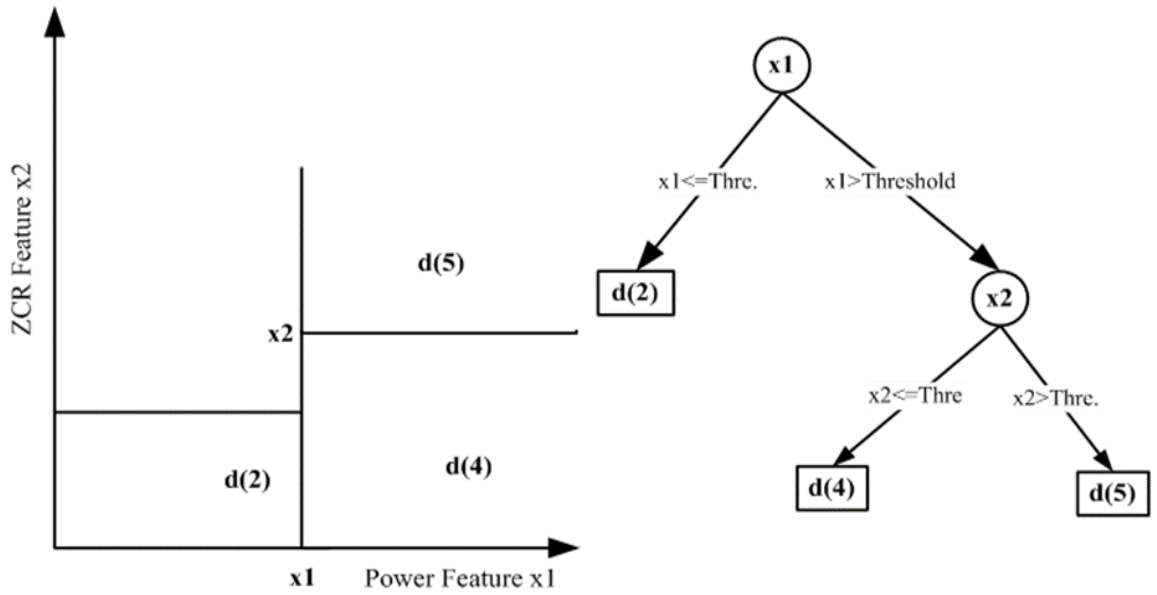


Figure 4-2 the determination of hyperplanes

4.3 Random Forests

Random Forests is based on ensembles of slightly dissimilar trees resulting from training on random training subsets. Originally, an ensemble methodology served to consolidate a set of existing modules, each module working on the same classification problem, in order to achieve a better global goal, with a lower error rate, more reliable results and better noise insensitivity than could be gained from a solitary module (Maimon and Rokach, 2008, Breiman, 2001a). Hence, the building of an ensemble classification tree using a random vector of the variables and then voting for the most popular class has resulted in a significant improvement in reducing the estimated error Breiman (1996a). An early experiment in random vector selection was by Breiman (1996b). Later on a number of studies have applied RFs and reported significant estimation error. Dietterich (2000) compared three methods for constructing ensembles of decision trees (namely Bagging, Boosting, and Randomization), and observed that randomising vectors give

better results than bagging. Consequently, random forests machine learning outperformed other machine learning, and this is confirmed by other projects carried out at Salford University by Al-Maathidi et al. (2015).

The tested vector passes through all the trees in the random forest, and each tree makes an independent decision. The final decision will be the class which has the most votes overall of the trees in the random forest, i.e., if the random forest consists of four trees used to classify an object into three classes, the prediction will be for the class which has the highest number of votes over all four trees. Figure 4-3 shows a simple RFs with B-trees.

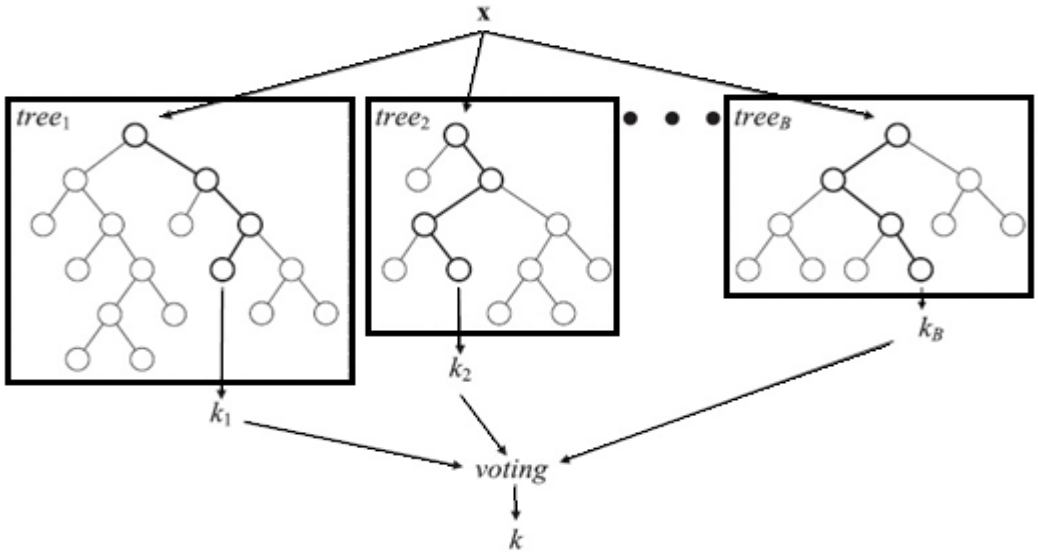


Figure 4-3 Simple Decision Tree Architecture with B-trees, k_i represents the probability

4.3.1 Random Forest Training

The construction of DTs starts at the root node then continuously partitions the feature space. A classification tree is established consuming a labelled data, $\{v = (\mathbf{x_i}, \mathbf{y_i}),\ i = 1...m\}$ where $\mathbf{x_i}$ is the data samples and $\mathbf{y_i}$ the respective class labels. The trees are growing based on the following strategy (Hastie et al., 2005):

- The Number of trees (B) is determined by the user at the beginning. Each tree is

trained independently of the others with a randomly selected subset of the training set per tree.

- The Out-of-bag (OOB) technique is employed for selection of the training samples, the training data set is divided into three parts, two of which are used for tree construction with the ability to swap, and the last one is left "out of bag" for testing. It is possible that some of the samples might be replicated in more than one tree and other of the samples never picked, as shown in Figure 4-4. Each tree in the RFs randomly selects a set of features and training samples from the training vector. OOB is used to calculate both the estimated error of the forests and the variable importance. This property makes RFs more robust against noise and the dependency on dataset problem.

- A binary split function $Sp = \{x, \theta_j\} \in \{0,1\}$ which is associated with each internal node, passes the patterns x at internal node j to either the left or right child node based on the decision (0 or 1).

This procedure is repeated for all features to select the best one to split. Then parameters θ_j are optimised for all tree nodes during training, to select features with a higher information gain (impurity function).

In general, splitting refers to determining which feature should be used and at which node by measuring impurity gain, and the optimal cut-point for that feature. The feature with the highest information gain value is selected to behave as a root node, which is used to split the dataset. In other words, features are organised based on the priority (importance) from the top down.

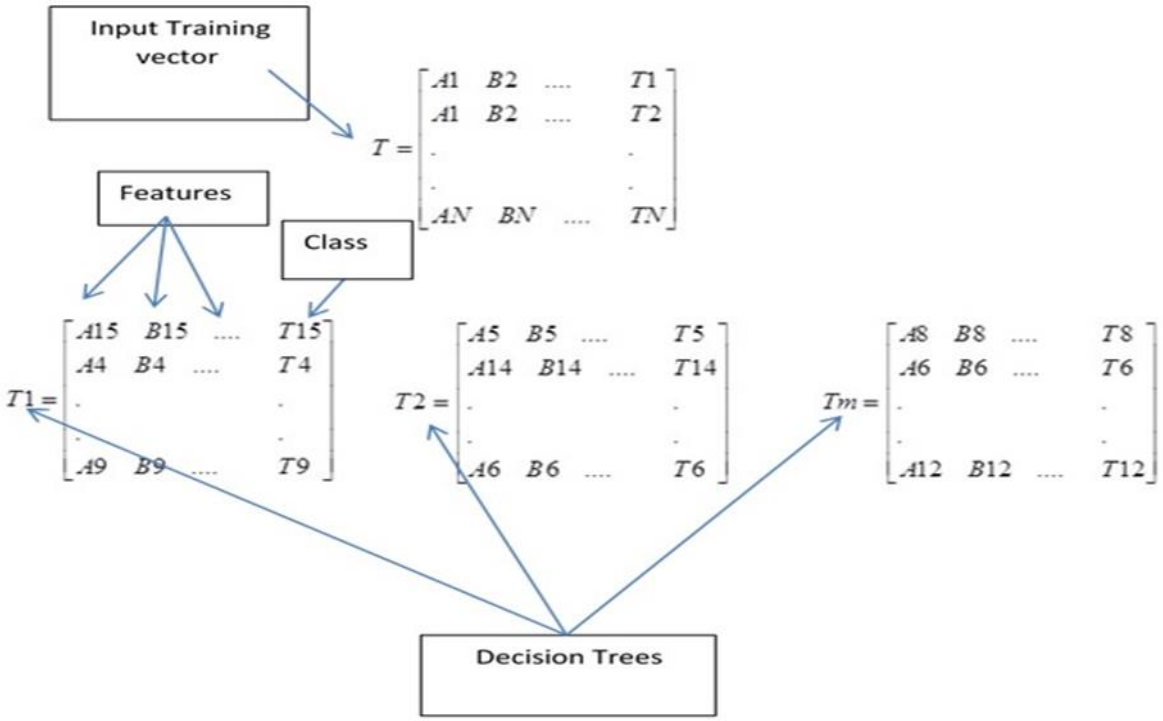


Figure 4-4 Subset samples selection (RF)

4.3.2 Impurity Function

The information gain achieved by a split represents the difference value between the impurity of the parent node and the summation of impurity for the two child nodes. Three different measure functions, the Gini, Entropy, and the misclassification rate, are widely used in the literature to measure the impurity in each node. In this study, the Shannon Entropy is used because it has been shown an ability to characterize the complexity of the signal contents in many publications. To name some, it has been used for measure the complexity of birds chirping by Briefer (2010) and Wang (2013), which has similar characteristics to the mixed soundtracks; the results were promising. Quinlan (1986) described the feature importance (gain) calculation using entropy and through the following suggestions.

- If there are two classes (*W* and *T*), then the probability that an object is (*W)* is (*w/w+t*) and the probability of its being (*T)* is (*t/w+t*), where *w* represents the

number of objects that belong to class (*W)* in that node and *t* is the number of objects belonging to class (*T)*.

- The information gain is calculated as impurity difference (reduction) between the impurity of parent node i^{th} and the summation of the impurities of the emitted nodes (left and right). Therefore, the total gain for (A) is given by Equation 4-1 Quinlan (1986):

$$gain(A) = I(w,t) - E(A) \qquad 4\text{-}1$$

The symbol *I* represents the entropy of the node i^{th} and it is calculated by Equation 4-2.

$$I(w,t) = -\frac{w}{t}\log_2\frac{w}{w+t} - \frac{t}{w+t}\log_2\frac{t}{w+t} \qquad 4\text{-}2$$

whilst *E(A)* represents the summation of the entropy of all emanating nodes from the i^{th} node, given by Equation 4-3.

$$E(A) = \sum_{i=1}^{v}\frac{w_i + t_i}{w+t} I(w_i, t_i) \qquad 4\text{-}3$$

where *v* is the number of potential child nodes using an attribute's cases index.

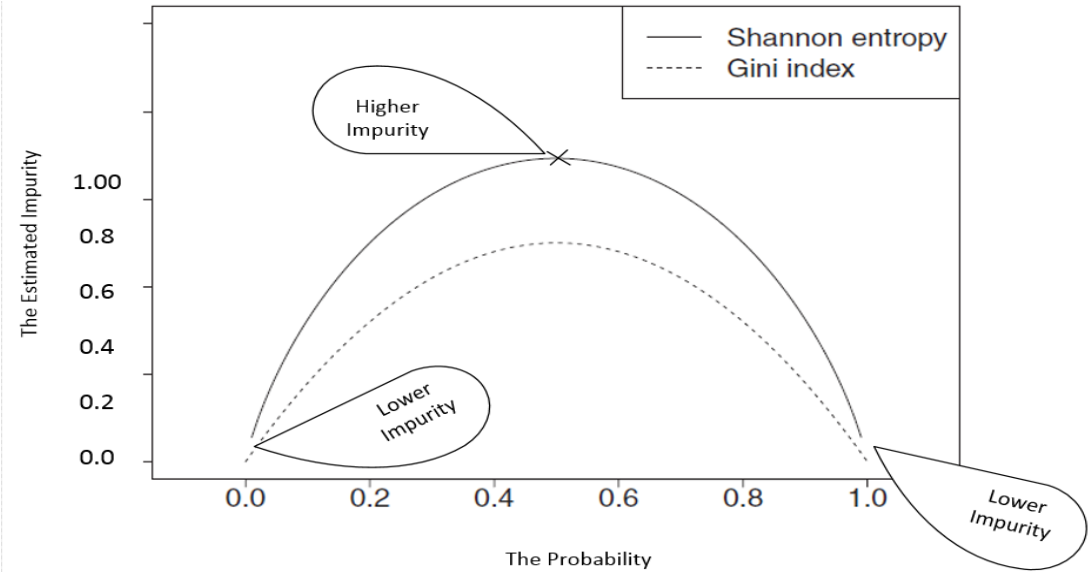


Figure 4-5 Impurity Estimation Functions

As shown in Figure 4-5, both impurity measure functions have in common that they touch their minimum for pure nodes with the single class being zero on Y-axis and their maximum for an equal mixture with the same relative frequencies for both classes.

4.3.3 Stopping Criteria

Many stopping rules have been proposed and developed in the literature. One of them is to carry on in the splitting process until all terminal nodes represent the pure class membership (0 impurity). This splitting method has a high probability of finishing up with a large tree that presents the data over-fitting problem and reflects a poor performance on an unseen test set because it is not generalized. Even though Breiman (2001b) was stated that RFs do not over-fit and this study has been cited in plentiful publications. This seems to suggest that this conclusion is based on a small size of actual dataset (a few number of real samples). Hence, this investigated dataset by Breiman (2001b) works to improve the behaviour that RFs would not over-fit. Later on Segal (2004) heavily argued that RFs do not over-fit. In addition, Luellen et al. (2005) stated RFs could over-fit when a large number of trees are utilised to construct the ensemble classifiers. Strobl et al. (2009) reported, "Other tuning parameters may be responsible for overfitting in random forests". Consequently, to avoid the overfitting problem, it has been suggested by some authors to stop when: a) a pre-set threshold for the maximum number of nodes is reached, b) a given threshold for the maximum depth of the node is reached or c) the impurity value reaches a minimum value (growing up is no longer significant). In this study, the applied stopping rule in order not to over-fit is to decide to stop if the probability is 95% sure about the class label in that node (the impurity reaches 0.05 or less, which means that the growing up of tree is no longer significant).

Finally, all the predictions from all the trees are combined together to produce a single

prediction that denotes the RFs output, and this is defined by Equation 4-4:

$$Pr(x/c) = \frac{1}{B}\sum_{i=1}^{B} Pr_i(x,c) \qquad 4\text{-}4$$

To conclude, the features of the RF technique could be summarised by:

- Random forests, which combine all patterns identified by a large set of single trees, can serve as a more flexible means for approximating different functional forms.
- The smoothing of hard decision boundaries also makes random forest trees more flexible than single trees in approximating functional forms that are smooth.
- Handling the missing value: observations that have missing values in the feature which is presently weighed, are neglected in the calculation of the impurity reduction for this feature. Conversely, the same observations, which have missing values, are involved in all other calculations, so that the RFs does not ignore the observations of attributes with missing values that can result in substantial loss in the statistics.

The key motivates the application of RFs, since it represents an intrinsically incorporate the redundancy included in the audio features. To make it clearer, literature reviews have indicated that there were not many researchers who have used the automated audio features extraction such as the optimisation and the machine learning. Hence, in the area of audio classification or categorisation, the utilised audio features (included low-level descriptors) were heuristically developed. All the related research is deeply influenced by psychoacoustic (sound perception and How are perceived by the human). Consequently, many features, which are sent to the machine learning, have the redundant information included in it. For example, there is defiantly some overlapping information

between MFCC and spectral centroid. In the RFs induction through the training phase, DT starts with the impurity assumption and try to end up with the pure probability. This is represented by the impurity function as a core internal function in RFs. Hence, DTs represent features with the redundancy and the impurity and works on minimising the impurity to get the overlapped thing separated. It is worthwhile to note once again that the audio feature extraction is based upon perception, which makes these features to have redundant information, and these DTs present this perfectly. Also, from empirical results of another researcher which are mentioned in Chapter 2, it seems as though DTs family is working better for this kind of similar type of research task. Finally, apart from Al-Maathidi (2015), RFs technique has not been very well reported for the high-level audio segmentation.

4.4 Summary

In the present chapter, a general illustrating of the Decision Tree technique in general and more specifically of the Random Forests technique which has been used to develop the non-exclusive classification that is highlighted in Chapter 6 as a solution for over-lapped content detection is provided. Furthermore, justification of the main utilised parameters such as impurity function, stopping criterion and number of trees is explained.

The provided information in this chapter will be a supportive presentation of knowledge to provide a basic understanding of machine techniques that are used in the system proposed in the subsequent chapters.

5 SINGULAR SPECTRUM ANALYSIS METHODOLOGY

5.1 Introduction

The architecture of a logical audio classification system and feature extraction process have been produced and applied as shown in the previous chapters. However, it remains difficult to resolve the harmonic overlap problem in mixed signals through logical classification and do straightforward filtering without the distortion that will be inevitably introduced in the separated signals after straightforward filtering processing. Therefore, to improve the performance of overlapped soundtrack classification, decomposing the overlapped oscillations into a number of oscillations with a lower ratio of overlapping and then classifying them separately is the key to success in the foregoing target. This might be done by exploiting the singular value decomposition technique and wide frequency range of mixed samples to generate clusters of oscillation.

This motivates the application of singular spectrum analysis, since it represents a statistical method (non-parametric) that is usually applied with arbitrary statistical signals, regardless of their distribution or processes, e.g. Gaussian or non-Gaussian, stationary or non-stationary. Singular Spectrum Analysis (SSA) is a method for time series decomposition and can be efficiently used to decompose signals to categorise the oscillation signatures (the patterns that appear in the lagged covariance matrix, see Section 8.3.2) of the time series over time (Fukunaga, 1970). Another motivation key that is the employed method need to be insensitive to the dynamical variation through the time period under test, since the dynamics of time series has frequently changed or gone through structural adjustments.

The expression singular spectrum derives from the spectral (eigenvalue) decomposition

of a two-dimensional matrix *A* into its combination of eigenvalues (spectrum). The identified eigenvalues (λ) for *A* are the non-negative numbers that make the matrix $A - \lambda I$ singular, *I* represents the identity matrix, see p. vii of Elsner and Tsonis (2013), for more details.

Every time series can be decomposed, using SSA into a series of matrices after mapping it into trajectory matrix and then processing it with the Singular Value Decomposition (SVD) technique (Section 8.3.3); each of these matrices shows glimpses of a particular signature of oscillation patterns. Following the grouping criterion, these matrices could be grouped into a number of smaller groups through summation operation, which is mapped onto the time domain with further processing. Each group should catch harmonic oscillation components. Sanie et al. (2015) give a clear example of the decomposition of the time structure S_{Nt} into noisy signal E_{Nt} and filtered signal C_{Nt}, as shown in Equation 8-1 (Sanie et al., 2015).

$$S_{N_t} = C_{N_t} + E_{N_t} \tag{5-1}$$

Most commonly, SSA works by decomposing data into the oscillation components such as noise and trend (data of interest) by deploying the SVD. SSA has been successfully used to separate desired signals and noises, with some advantages of lower distortion being imposed on signals. The SSA approach is applied to EEG and ECG signal processing for separating and localising a combination of signals produced from frequencies/amplitudes that are generally different (Bonizzi, et al., 2015; Sanei and Hosseini-Yazdi, 2011; Wang, Liu, and Dong, 2016).

5.2 Basic Methodology of SSA

The procedures of SSA comprise four main parts. First, embedding: a given signal vector is transformed into two a dimensional matrix which is called the Trajectory Matrix

(TM). Then, the lagged covariance matrix of the embedded matrix (TM) is calculated. Secondly, Singular Value Decomposition; this stage holds the most significant two steps in the SSA technique. It consists of calculating two matrices of SVD. The first matrix is a square matrix with diagonal non-negative values representing eigenvalues and the second matrix represents left eigenvectors, each vector denoted by one column. Then, these two matrices with the TM can be used to compute the Principal Components (PCs).

The reconstruction stage includes three more steps, the grouping step which is almost complete based on a scree plot (demonstrates the eigenvalues in descending order versus the index of the eigenvalues on the X-axis) to determine the sufficient eigenvalues. Then, the corresponding Principal Components (PCs) to these sufficient eigenvalues are added together, and other, undesired PCs are omitted.

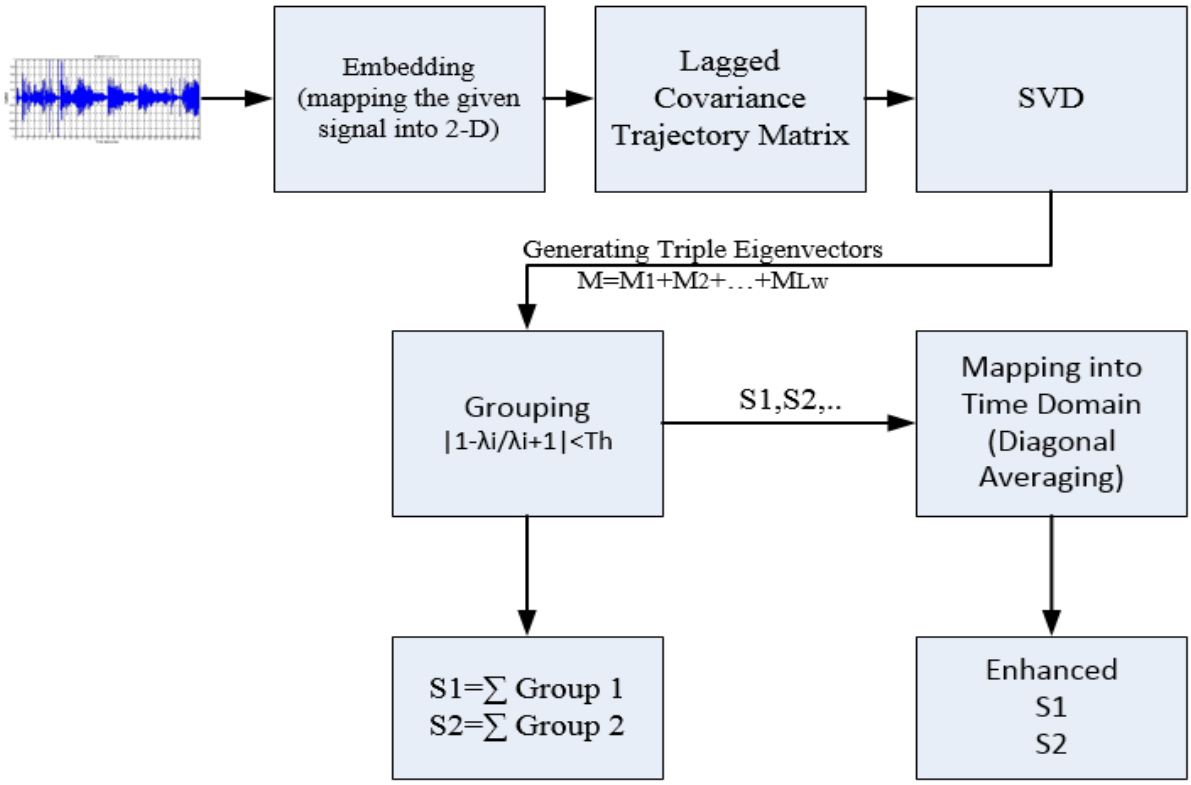


Figure 5-1 Singular Spectrum Analysis Algorithm

Finally, the filtered target signal is mapped back onto the time domain represented by the vector through diagonal averaging. The general SSA algorithm is illustrated in Figure 5-1.

5.2.1 Embedding

To make a multivariate statistical analysis from a univariate time record possible, a TM transformation method needs to be deployed. The idea of TM is related to chaos theory (Ruelle, 1984). Ruelle states that "we shall admit that the parameters specifying the system at time t+1 are given functions of the parameters at time 1" (Ruelle, 1984). In summary, by using lagged time of a single time series, the coordinates of the dynamic system can be defined. The number of lags K is called the embedding dimension. The time L_w spanned by each embedding vector is often known as the "window length" of the embedding dimension. The number of embedding vectors K can be calculated by the following formula (Ruelle, 1984).

$$K = N_t - L_w + 1 \qquad 5\text{-}2$$

where N_t denotes the length of the time series that is being processed with SSA. The same principles (lagged time and embedding dimension) are applied for the purpose of singular spectrum analysis, where the delays represent the transformation of a vector of time records into a multivariate set of time observations (Elsner, 2013, p. 39). The processing time of SSA very much depends on the window length of the embedding dimension L_w, and the length of the time series under test. These two factors lead to an increase in the dimensionality of the Trajectory Matrix (TM), thus increasing the overall computation processing time.

In order to minimise the processing time of SSA in this study, the soundtrack samples are processed frame by frame instead of processing the whole audio file directly. Thus,

the TM becomes much smaller due to selecting a smaller statistical dimension (embedding dimension) as well. Then, the given frame, $f(x) = \{x_1 \ . \ . \ x_L\}$ with a time series of size L_t, is mapped using the lagged time principle onto a two-dimensional embedded time series as shown in Equation 5-3 (Elsner, 2013, p. 39).

$$Y = \begin{Bmatrix} x_1 & x_3 & \dots & x_K \\ x_2 & x_3 & \dots & x_{K+1} \\ \vdots & \vdots & \dots & \vdots \\ x_{L_w} & x_{L_w+1} & \dots & x_{N_t} \end{Bmatrix} \qquad 5\text{-}3$$

where the first column vector denoting the system at lag time $(i\text{-}1)$ with length (L_w), and K represents the number of these row vectors; the latter depends on the embedding dimension size L_w. K can be calculated by Equation 8-2 as shown previously. L_w is $(2 \leq L_w \leq N_t - 1)$. Each column vector $\mathbf{y}_i = \{x_i \dots, x_{i+L_w-1}\}, i = 1,\dots,K$, where x_i reflects the time domain samples. The successive vector $\mathbf{y_i}$ should be long enough to characterise the dynamic of the discrete time series (Golyandina et al., 2016; Sanei et al., 2015). The shifting increment is one sample at a time. **Y** at lag 0 represents the first L_w elements from time series vector, $lag1 = \{x_2,\dots,x_{2+L_w-1}\}$ as depicted in Figure 5-2.

As before, SSA algorithm performance is highly dependent upon the selection of the sliding window length L_w. Rukhin mentioned that the size of L_w should be long enough to represent significantly separated components but not longer than $N_t/2$ (Rukhin, 2002), whereas Harris et al. (2010) reported that it needs to be greater than $N_t/2$, when $K > L_w$ (Y "has many rows larger than columns").

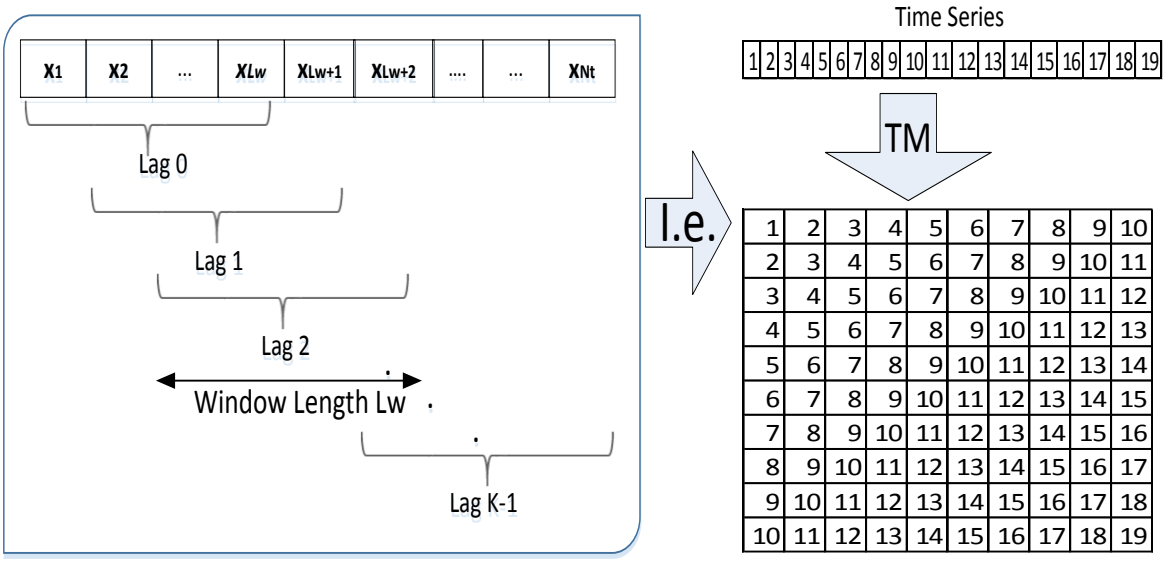


Figure 5-2 Trajectory Matrix Production

The structured properties lead to a number of filtering interpretations. The same assumption was made by Alexandrov et al. (2008) who used SSA for decomposition of specific protein type profiles into the sum of a signal and noise; they concluded that even where the size of N_t was small (for substantial noise or high-rank r) and separability was insufficient, small L_w can be used to extract the trend (data of interest). This technological concept in SSA describes how well different constituents can be split from each other. It is worth noting that the singular spectrum analysis decomposition stage delivers significant results if the resulting additive ingredients of the series are approximately separable from each other (Golyandina et al. 2002).

It is worthwhile also to note that the size of the window depends on several criteria such as the aim of the analysis/prediction and the forecasting horizon. Insignificant decomposition and inferior prediction results are highly related to improper selection of the window size. Since the objective of this research is filtering some of the speech and music components not hitherto examined with SSA, to determine the optimum length of L_w more investigation has been carried out for selection of the optimal size following

the heuristic method, as illustrated in the next chapter.

5.2.2 Lagged-Covariance Matrix

The second stage in the decomposition phase represents computing the lagged covariance matrix. This can be calculated through measuring the relation between the elements of the TM. Where the element in the (i, j) position of the covariance matrix represents the covariance between the (i^{th}, j^{th}) and (j^{th}, i^{th}) elements of the TM, Elsner et al. (2013) illustrated that the repeating patterns reflect the oscillation in the time series. The lagged autocorrelation matrix is considered one of the Fourier analysis methods, as highlighted by Blackman and Tukey (1959). Moreover, these oscillations can be calculated as the product of the TM and its transpose (for more information see Section 4.2 of Elsner et al, 2013), which is represented by the covariance matrix **Cx** as shown in Equation 5-4.

$$\mathbf{Cx} = \mathbf{Y}\mathbf{Y}^T \qquad 5\text{-}4$$

where T denotes the transpose of TM ($\mathbf{Y}$).

Later, Broomhead and King (1986) suggested normalising the calculation in Equation 8-4 by the window length of the embedding dimension L_w, giving what can be considered to be an estimation of the lagged covariance matrix. In general, the elements of **Cx** represent the autocorrelation between all possible pairs in the pattern that appears in the K-windows. It is worth noting here that **Cx** ($L_w \times L_w$) is a real and symmetric matrix.

5.2.3 Singular Value Decomposition (SVD)

Every matrix can be factored into three pieces as presented in Equation 8-5 (Elsner et al, 2013).

$$\mathbf{Y} = UDE^T \qquad 5\text{-}5$$

where *E* accounts for the left eigenvectors (extracted from of $\mathbf{YY}^{\mathrm{T}}$), and **Y** denotes TM in this example. Meanwhile, *D* is a diagonal matrix of singular values and *U* is an orthogonal matrix $(L_w \times L_w)$, which represents the right eigenvectors (the vectors that are extracted from $\mathbf{Y}^{\mathrm{T}}\mathbf{Y}$.

Therefore, we have an orthogonal, diagonal, and inverse orthogonal or physically (rotation, scaling, and then reverse rotation) components. Thus, Equation 5-4 of covariance matrix calculation can be written as Equation 5-6 as follows.

$$\mathbf{Cx} = \left(UDE^T\right)^T \left(UDE^T\right) = EDU^T UDE^T \qquad 5\text{-}6$$

Since $U^{\mathrm{T}}\, U$=I=1, then Equation 8-7 can be written as:

$$\mathbf{Cx} = ED^2E^T \qquad 5\text{-}7$$

Consequently, $D^2 = \Lambda$, Λ being a diagonal matrix (lambda) whose entire non-negative element represents the eigenvalues of the covariance matrix **Cx**.

Furthermore, if $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_d > \lambda_{d+1} = \lambda_{d+2} = \cdots \lambda_{L_w} = 0$ are the ordered values of the diagonal matrix Λ, hence $\sqrt{\lambda}_1, \sqrt{\lambda}_2, \sqrt{\lambda}_3, \ldots, \sqrt{\lambda}_{L_w}$ are the singular values of **Y**. The eigenvalues can be computed from Equation 5-8

$$\left(\mathbf{Cx} - \lambda I\right) = 0 \qquad 5\text{-}8$$

where *I* here is an identity matrix that can be defined as a square matrix with ones on the main diagonal and zeros elsewhere. Now, for each extracted eigenvalue a corresponding eigenvector can be calculated through Equation 5-9 (Elsner et al, 2013)

$$E = (\mathbf{Cx} - \lambda I) = 0 \qquad 5\text{-}9$$

The eigenvector matrix with the dimension of $(L_w \times L_w)$ of the matrix **Cx** reflects the

temporal covariance of the time series, at different lags, as explained above. The extracted eigenvectors are considered the axes of the new coordinate system. Therefore, any scalar multiple of these vectors is also can be considered an eigenvector of the given matrix. Strictly speaking, $diag(\Lambda) = \lambda_1^2, \lambda_2^2, \lambda_3^2, \ldots, \lambda_{L_w}^2$ is the diagonal matrix of eigenvalues of **Cx** and $E = \left(\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_{L_w}\right)$ is the corresponding orthogonal matrix of eigenvectors of **Cx**, where each $\mathbf{e_i}$ represents a single eigenvector with length L_w.

After calculation of λ and E, the Principal Components (PCs) of the time series can then be constructed. The PCs are again time series, of the same length as the K value. The only difference with the TM is that each point is represented by a different coordinate (Claessen and Groth, 2002). Drawing on Pearson (1901) the PCs can be computed by Equation 5-10.

$$\mathbf{v}_i = \frac{\mathbf{Y}^T.\mathbf{e}_i}{\sqrt{\lambda_i}}, i = 1, \ldots, L_w \qquad 5\text{-}10$$

v will be a ($K \times L_w$) dimensional matrix and each column represents an individual PC vector. The variance of each consecutive PC is equal to the eigenvalue of the corresponding eigenvector, i.e. the difference between the first and second PC ($\mathbf{v}_1$ and $\mathbf{v}_2$) is identical to the difference between the first and second eigenvalues (λ_1 and λ_2). The PC matrix is again in the time domain and each individual PC contains a part of the oscillation information and can be isolated and investigated independently from other PCs due to their orthogonal characteristics (Claessen and Groth, 2002; Elsner and Tsonis, 2013). The SVD of the TM can be written as shown in Equation 5-11 (Elsner et al, 2013)

$$\mathbf{M} = \mathbf{M_1} + \mathbf{M_2} + \ldots + \mathbf{M_d} \qquad 5\text{-}11$$

where $\mathbf{M_i}$ denotes elementary matrices or, as they are sometimes called, reconstructed components that can be computed as the sum of the rank-one bi-orthogonal decomposition of the trajectory matrix TM into its orthogonal bases. Each elementary matrix can be expressed as in Equation 5-12

$$\mathbf{M}_i = \sqrt{\lambda_i}\,\mathbf{e}_i \mathbf{v}_i^T \qquad 5\text{-}12$$

Based on the two preceding paragraphs, any time record can be filtered by eliminating any unwanted oscillation patterns through selecting their corresponding PCs. The determination of unwanted PCs is done usually based on the eigenvalue distribution.

The approach in this study is the applying of SSA as a well-known method for frequency decomposition, as a filter technique before the machine learning stage to refine the classification results. This is done through pruning the harmonic frequencies of a given signal to keep the components of only one class and eliminate all other components.

5.2.4 Grouping

As mentioned above, each one of the elementary matrices, generated by convolution* of one of the PCs with its corresponding eigenvector and weighted by eigenvalue, represents a particular oscillation of the signal. Often eigenvalue distribution is used for boundary detection of the subspace belonging to the signals of concern because the distribution of eigenvalues is calculated based on the variance. For example, in the noise case, the noise variance is represented by the higher space eigenvalues, since the noise is autocorrelated (having higher associated frequencies) Elsner and Tsonis (2013, p. 99), project onto the higher subspace which represents lower variances in the frequencies. In contrast, the higher variances in the frequencies are projected on the lower subspace of the eigenvalue distribution. The line of eigenvalue distribution descends with

increasing index (higher associated frequencies). Hence, in the case of noise cancellation, the high order eigenvalues are almost isolated due to their representing low oscillations, and the detection of the isolated boundaries is almost complete based on the analysis of the scree plot of the eigenvalues. Usually, the low order eigenvalues, which are related to the desired components, are determined by the standard method. This is done by selecting the first *ds* matrices whose corresponding eigenvalues achieve the following condition.

$$\frac{\sum_{i=1}^{ds} \lambda_i}{\sum_{i=1}^{L_w} \lambda_i} \geq th \qquad 5\text{-}13$$

Many authors have defined *th* to be equal or higher than 0.85, for example Ghaderi et al. (2011), Mohammadi et al. (2016), Mamou and Feleppa (2007). This ratio is identified to reject the components most likely to correspond to the floor of the noise subspace. The desired elementary matrices in the first group $\mathbf{I}_1 = \sum_{i=1}^{ds} \mathbf{M}_i$, are aggregated together to implement each group as one matrix, of the same dimension as TM (**Y**)**,** whilst leaving out undesired components in the second group that reflect the noise components, $\mathbf{I}_2 = \sum_{i=ds+1}^{L_w} \mathbf{M}_i$.

The effectiveness of SSA depends significantly upon the successful analysis of the eigenvalues and selection of appropriate groups through convenient criterion to reconstruct the desired components. Each matrix in the selected group is supposed to have similar harmonic characteristics (Hassani, 2010). Previous studies have reported differ-

ent criteria for detection of the boundaries of the groups. Jarchi and Yang (2013) reconstructed only the second and third elementary matrices to recognise walking patterns - walking downstairs, level walking, and walking upstairs - using SSA instead of traditional classification techniques (Jarchi and Yang, 2013). They mention that the second and third elementary matrices are related to the dominant oscillation of the acceleration signal if the signal has a periodic pattern. Also, their corresponding λ_2, λ_3 will be generally similar. Vautard and Ghil (1989) also have stated that each pair of eigenvalues has a similarly equal value, which means this pair corresponds to a significant oscillation pattern. Mohammadi et al (2016), as explained earlier, extracted brain waves, sleep spindles, and K-complexes from a sleeping EEG signal. They adopted new separation criterion based on their sleep signal analysis. The signal is factorised into eigenvalue pairs: only eigenvalues with nearly similar values, which thus seem to be pairs, are selected. The variances between successive eigenvalues are computed first. Then, the eigenvalues which have the smallest difference are selected. Finally, the following condition (Equation 5-14) is applied by the author to recognize the components. Their eigenvalues are within the range (Mohammadi et al, 2016)

$$1-\frac{\lambda_i}{\lambda_j} \leq th \qquad 5\text{-}14$$

where λ_i, λ_j represent the eigenvalues which have nearly similar values and the value of *th* is changed with regards to the pattern amplitude. Thus, the authors set a particular *th* value for each PC. The higher threshold value is selected for alpha, theta, and delta because they have higher amplitudes than spindles. Meanwhile, because the spindles are represented by lower amplitudes, a lower value for *th* is selected (Mohammadi et al.,2016). Ma et al. (2012) combined SSA with the BSS technique to separate the mixed signal into two components. Initially, SSA is used to decompose the signal into two

groups through calculating the contribution of each elementary matrix $\mathbf{M}_i$. The contribution of matrix $\mathbf{M}_i$ is calculated using the share of corresponding λ_i by the formula 5-15 (Ma et al., 2012).

$$\mathbf{r}_i = \frac{\lambda_i}{\sum_{j}^{L_w} \lambda_j} \qquad 5\text{-}15$$

Hence, each $\mathbf{r_i}$ reflects the contribution of corresponding elementary matrices. Then, the corresponding PCs that have the greatest contributions are summed together to generate the first group, and the remaining PCs make the second group.

5.2.5 Diagonal Averaging (Reconstruction of the one-dimensional series)

Finally, after determining the boundaries of each group, the matrices in each cluster are summed together. The additive matrix **I** for each group is transformed back to one-dimensional time domain vectors through diagonal averaging. Changeover back to the univariate time series can be accomplished by averaging over the diagonals of the matrix **I**. Then, each of these output vectors will tend to characterise a particular signal component.

It is worth noting that the group matrices are not Hankel matrices, which means that all the elements along the diagonal $i+j = const$ are not equal. Therefore, there is a need to perform diagonal averaging over the diagonals $i+j=const$ to reconstruct the signal. This corresponds to averaging the matrix elements using the rule shown in the Equation 5-16:

$$f(k) = \mu\big(I(i,j)\big), \forall i + j = k+1, k = 0,1,...,\arg\min(K, L_w) \qquad 5\text{-}16$$

where μ represents the statistical mean. Figure 5-3 illustrates the diagonal averaging

process.

2	1	7	0	2	7	7	4	0	0
8	9	2	9	4	0	7	9	4	5
1	9	1	7	6	2	9	6	1	0
2	4	2	4	6	3	8	6	9	8
1	1	3	5	3	6	3	8	7	8
2	2	4	2	3	1	6	8	5	7
4	4	5	4	9	7	1	5	4	1
3	5	0	9	0	1	0	1	0	6
9	2	2	5	8	6	7	2	6	5
4	6	8	5	9	4	5	8	0	9

Figure 5-3 Diagonal Averaging Method

5.3 Example of SSA

To make the methodology of singular spectrum analysis clearer, the following example of a corresponding periodic sinusoid signal with additive noise component is used. This was implemented in Matlab. The signal was denoted by the expression in 5-17.

$$x = \sin(K\pi F_r t) + A(N_t) \qquad 5\text{-}17$$

where K represents the cycle frequency over time t, time variable t is changed from 0 to .005 with step 0.001 whereas F_r denotes the frequency of each signal (1000), A is the amplitude of noise (0.2), and N_t is a random vector with a size similar to t vector length. The entire signal and its additive noise are depicted in Figure 5-4. Let us consider that the time series is represented by the vector $s(n) = \{x_1, x_2, \ldots, x_{Nt}\}$, where N_t denotes the length of the time series. Now, setting the embedding dimension with length equal to L_w, $L_w = N_t/2$, the number of column vectors K can be calculated from Equation 5-3. Then, the time series is transformed into a 2-dimensional trajectory matrix **Y** (L_w X K), using K and L_w parameters and TM as defined in Figure 5-2. The next step is the calculation of the covariance matrix **Cx,** where the diagonal vector of this matrix denotes the

variance of each column.

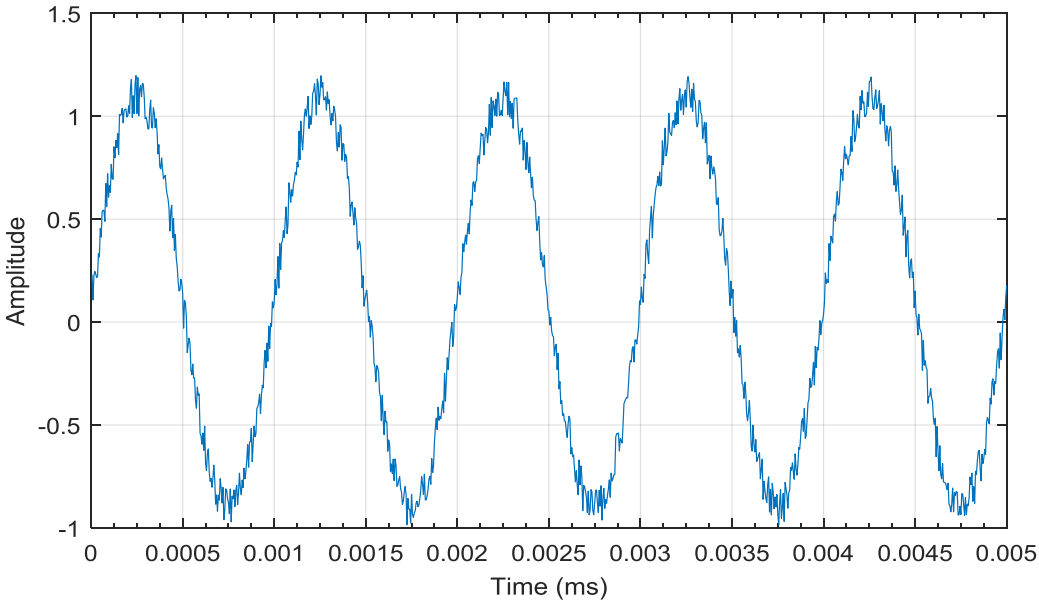


Figure 5-4 Input sinusoidal signal with additive noise

We continued employing **Cx** based on Equation 8-4 and normalised it to the embedding dimension length (see Section 5-3-2). Both the left eigenvectors (*E*) and eigenvalues (λ) of **Cx** are extracted using Equation 5-8 and Equation 5-9 respectively. The corresponding singular spectrum is illustrated in Figure 5-5 (the first 50 greatest λ).

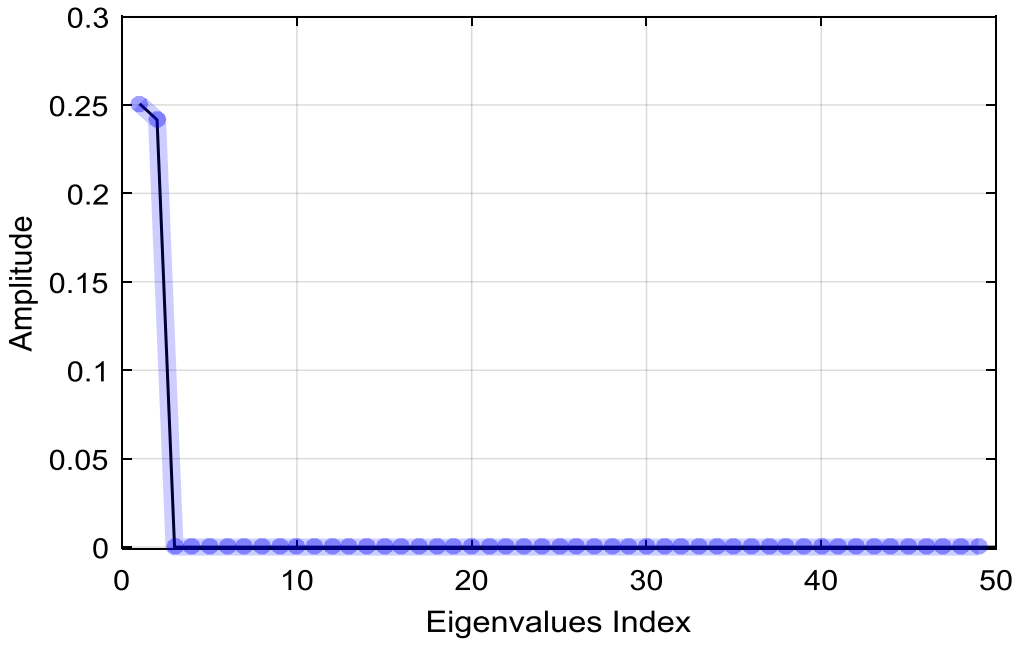


Figure 5-5 Singular Spectrum of time series under test

It can clearly be seen that the first two λ are significantly higher than all others and take more than 95% of the power of the signal under test, while all other components are near to zero. This means that the first two PCs give glimpses of the periodic components whereas all others catch random components.

Hence, the PCs using the eigenvalues, eigenvectors, and TM can be computed using Equation 5-10. Figure 5-6 depicts the first six principal components. It is apparent from the figure that the first two PCs contain practically all the variance of the time series; this is consistent with the first two λ values.

As illustrated above, each elementary matrix can be reconstructed by projecting the pre-calculated principal components back onto the eigenvector coordinates using Equation 5-12. Hence, the anticipated subspace belonging to the sinusoidal component and the undesired subspace corresponding to the additive noise are separated. The grouping criterion in this example depend on the λ distribution in the components of Figure 5-5.

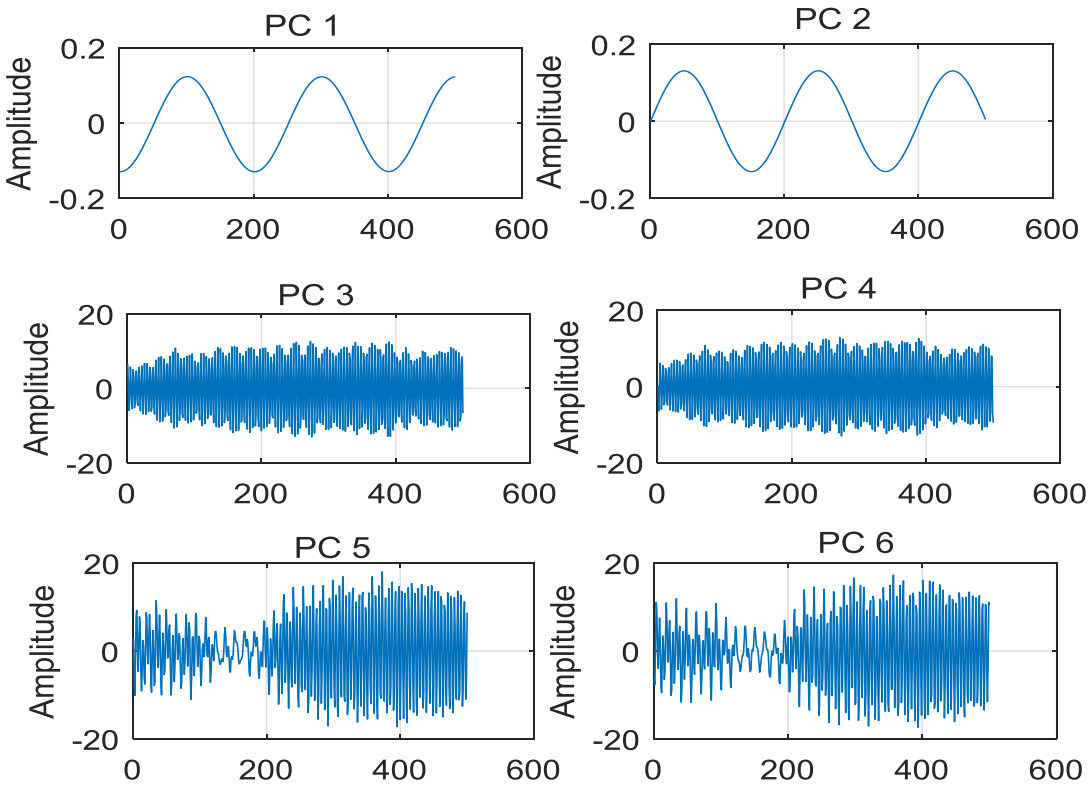


Figure 5-6 First 6 Principal Components of time series under test

In this step, the elementary matrices are divided into two groups, and then the matrices within each group are aggregated. The first group includes the first two elementary matrices, and the second group includes all others. Finally, diagonal averaging is performed to transform a matrix into a time series, as shown in Figure 5-7.

Comparing these two reconstructed signals easily leads to the conclusion that the components have been separated almost without distortion. In addition, by comparing Figure 5-7 to Figure 5-4 it is evident that the singular spectrum analysis has separated the mixed components accurately and that the singular spectrum analysis has performed the separation without any prior knowledge of the components themselves or the mixing process.

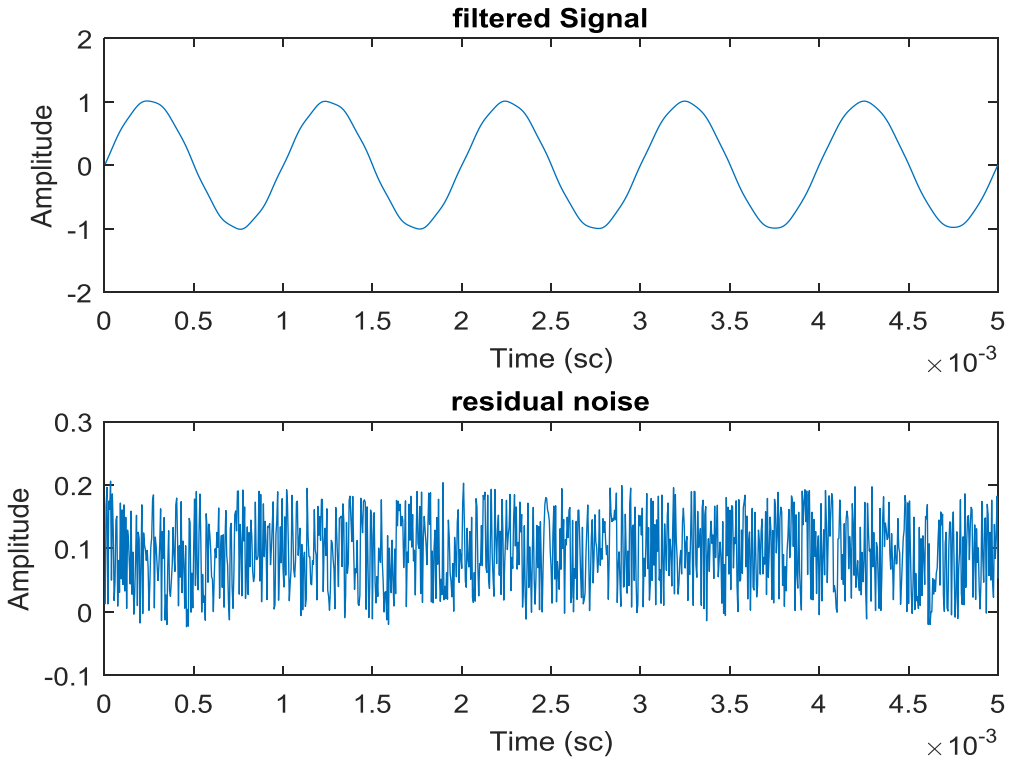


Figure 5-7 SSA demonstration of recovered source signal

This experiment provides a clear illustration of the usefulness of the technique in the face of the permutation uncertainties discussed above. The magnitudes of the sinusoidal signal in Figure 5-7 and noisy signal Figure 5-4 are small. This demonstrates that the separation achieved by singular spectrum analysis was efficient.

5.4 Summary

The rationale behind the singular spectrum analysis technique has deliberated. Also, the main steps and methodology for the geometrical concept of rotation based on higher variance are clearly illustrated. In addition, a summary of the literature regarding the use of SSA in different areas were briefly discussed. As a conclusion, the decomposition of pattern oscillation and Eigen-space analysis applied in the SSA method make it more robust than standard filters and the other BSS methods for pattern categorization and overlap mitigation. Chapter 9 will include a description of the designed method using SSA techniques to classify the overlapping soundtracks.

6 SAMPLE COLLECTION AND DATASET

6.1 Introduction

The scope of this chapter is to present an outline of the basics of the dataset, and review the benchmark and widespread GTAZAN dataset samples that are to be used in this study. Furthermore, it covers how the samples from this dataset have mixed to generate mixed soundtracks in a different mixing ratio. This chapter will be a supportive presentation of audio samples to provide a basic explanation of the mixed soundtracks that are used for evaluation of the suggested methods in the subsequent chapters.

6.2 Dataset

The speech and music samples which are used for the evaluation issue were from the GTZAN database (Tzanetakis, 2014). A benchmark database is essential for the study of speech and music classification. Such a "standard" dataset is specifically beneficial in this work. In addition, this will make comparisons with the results from others work more accurate. The GTZAN archive is normal choice due to the fact that it is considered the most popular database deployed and validated by many other researchers and cited in numerous publications in the speech/music discrimination and music categorizing areas. Some examples of which are, Barbedo and Tzanetakis (2010), Sturm (2012), Sattar et al. (2011), Tzanetakis (2005), Tzanetakis and Cook (1999a), Tzanetakis and Cook (2002), Huang et al. (2014), Zhang et al. (2015), Lu et al. (2016).

The GTZAN music/speech collection has been created by Tzanetakis and his supervisor Cook (2002) through his PhD project. It consists of 120 tracks, half of which are music and the other half speech, each track being 30 seconds long (120*30 s= 1-hour). The GTZAN dataset also includes ten music genres (Jazz, Classical, Pop, Rock, Hip-hop,

Country, Disco, Blues, Metal, and Reggae). Because the GTZAN is designed for MARSYAS software with emphasis on music information retrieval (MIR), some of the music samples contain singing content. Therefore, some effort has been made through listening to the entirely of the dataset many times, along with using automatic methods where possible to ensure that all music samples which are included in the training dataset are pure music without singing. In addition, all silent gaps included in the speech samples have been omitted.

As for the generalisation issue, though, the GTZAN database is adequate since it includes many different styles of speech and music samples. For example, the speech samples involve utterances by children, male, female, a speaker of different languages, loud speech, speech with laughter, etc. Also, almost all music genres are included, as explained before.

In some experiments (Chapter 7), the Audio and Acoustics Signal Processing challenge (AASP) dataset, which is published by IEEE and explained before in Section 2.4.3, was used. The AASP challenge provided two datasets: one for scene or soundscape classification and the other for event classification. The AASP data was gathered from 10 different places in the London area: inside an office, park, quiet street, open market, restaurant, supermarket, tube train, tube station, bus and busy street. The event dataset collected from inside the office is further divided into two sets: monophonic denoted as Office Live and polyphonic denoted as Office Synthetic. Event types used were alert (beep sound), clearing throat, cough, door slam, drawer, keyboard clicks, keys (keys put on the table), knock (door knock), laughter, mouse click, page turn, pen drop, phone, printer, speech, and switch

6.3 Mixer Model

An audio mixer that mixes pre-recorded samples according to their signal intensity has been developed using MATLAB code. Figure 6-1 shows the experiment flowchart, to mix speech and music signals with a predefined percentage.

The Speech/Music mixing strategy used in this study is empirically verified and published to reflect the best mixing for representing mixed soundtracks (Mohammed et al., 2015). The procedure followed for the mixed speech/music samples in different speech-music ratio can be reviewed as the following:

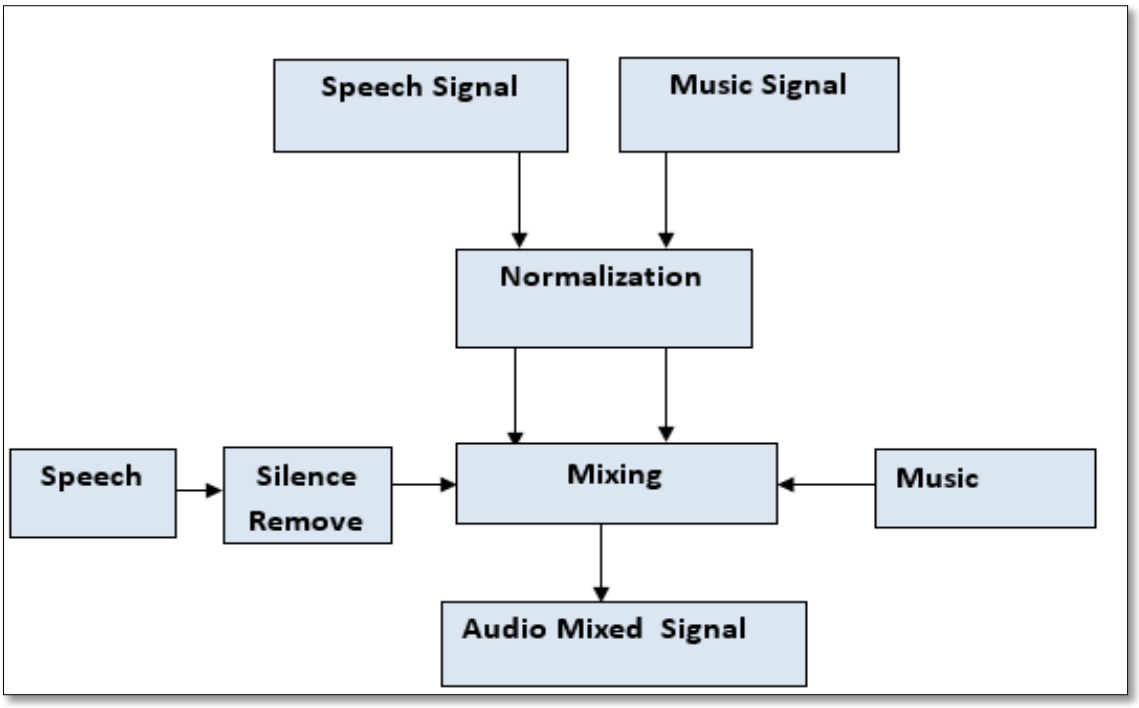


Figure 6-1 Speech Music Mixer Model

6.3.1 Normalisation Stage:

The issue of normalisation needs to be addressed, so that the music and speech signals can be added in the correct proportion so as to avoid misinterpretation. The default method is normalisation of the mixed or compared signals to the same perceived level and it is a significant factor for reliability that the input signals have the same level

(McKinney, 2008). The mixed signals are processed to have the same (RMS). The given audio signal can be normalised according to Equation 6-1.

$$\hat{x}(i) = x(i)En \tag{6-1}$$

$$En = \frac{\arg\max(Rs, Rm)}{\arg\min(Rs, Rm)} \tag{6-2}$$

En: the RMS normalisation factor, *Rs*: RMS of speech signal and *Rm*: RMS of music signal.

6.3.2 Mixing stage

In this experiment, 60 speech samples are mixed with 60 music samples at 9-difference speech-music ratios as shown in the Figure 6-2. The experiment mixes the speech and music samples in terms of the RMS speech to music ratios. The predetermined mixing ratios were chosen to examine how features are related with the signals' content ratio changing in order to optimise the feature selection algorithm. As shown in Figure 6-2, speech-music ratio values ranging from -20 to 20 dB in steps of five are used to mix speech with music samples. The goals from mixing audio experiment are as follows:

- Generate audio benchmark data.
- Set the limitations and evaluation of the classification systems.
- Evaluate and compare the suggested methods.

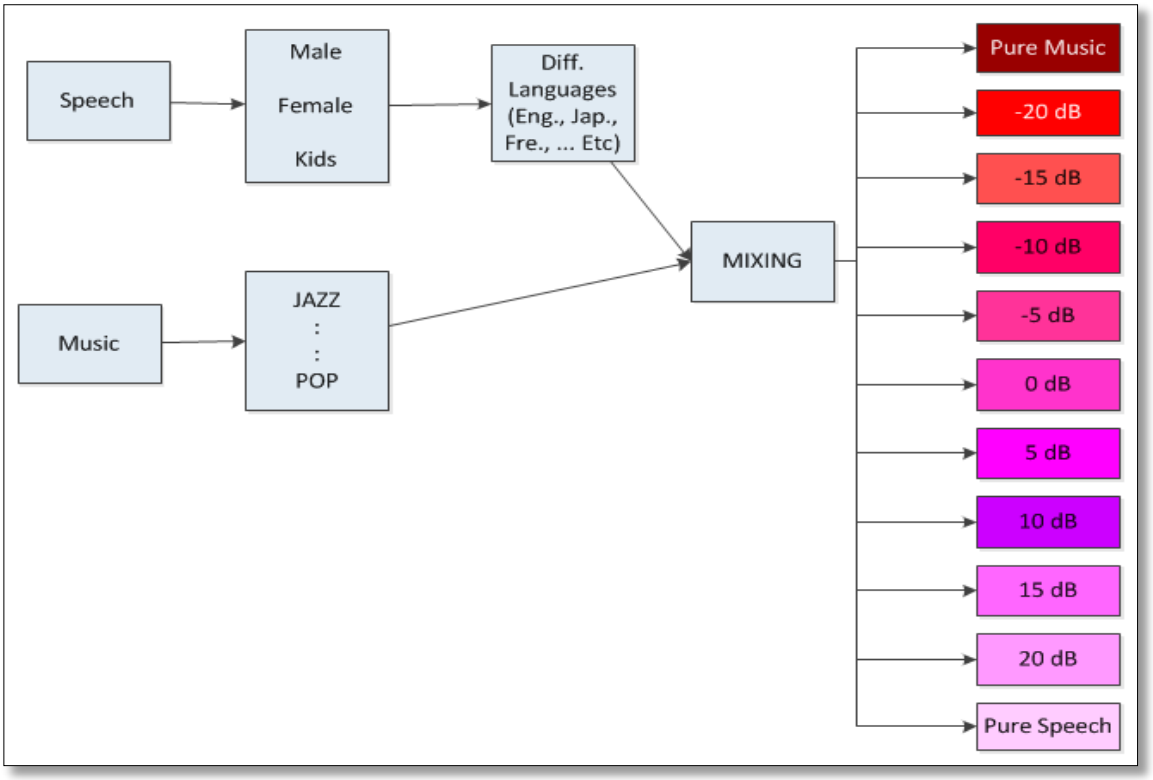


Figure 6-2 Mixing Architecture of Speech and Music Samples

6.4 Summary

In this chapter an outline of the basics of the dataset is demonstrated, and the benchmark and widespread GTAZAN dataset samples that are to be used in this study are reviewed. Furthermore, a mixer model that have mixed the samples from this dataset is suggested, developed and described. It covers how to generate mixed soundtracks in a different mixing ratio. The normalization method is addressed, so that the music and speech signals can be added in the correct proportion so as to avoid misinterpretation. The mathematical model that used in processing the mixed signals to have the same (RMS) is developed and presented. The generated and explained samples in this Chapter will be used for evaluation and validation of the proposed methods in this study in the subsequent Chapters.

7 EVALUATION AND EXTENSION OF EXISTING SYSTEM – A CASE STUDY

7.1 MARSYAS EVALUATION AND EXTENSION

7.1.1 Principles Framework of MARSYAS

MARSYAS is considered to be one of the well-known open source semantic audio analysis systems, evidenced by it is broad use and reuse in many renowned projects, including those conducted in national broadcasting houses. Moreover, it uses an effective audio classification pre-processor to categorise content, gating the music segments for information retrieval. It is an open source software framework, licensed by GNU General Public License, and developed using C-language and Python for audio processing with specific emphasis on Music Information Retrieval applications. MARSYAS started as the Ph.D. project of George Tzanetakis and his supervisor Perry Cook, and other participants from around the world (TZANETAKIS, 2002, Tzanetakis, 2009). It has employed established software as plug-ins for rapid prototyping of real-time classification of audio. Examples include VAMP and sonic visualizer (for audio feature extraction), QT (design interfaces), CMake (used to control the software compilation process), LATEX and MIKTEX (document design), Ghostscript (an interpreter for portable document format (PDF)), Doxygen (standard tools for generating documentation from annotated C++ sources) and a number of other pieces of software. Similar to many other audio analysis tools, MARSYAS used a machine-learning regime. At the beginning, a large labelled dataset is needed to train and validate the system. In the retrieval phase, the system is expected to perform as it was trained to do. Figure 7-1 MARSYAS system framework presents the general framework of the MARSYAS system. It contains five

levels. The given signal is portioned into segments and dimensionality reduced by calculating a number of common features. Then, in the memories level, both the variances and the means of features are calculated over a window of approximately 1-second window size.

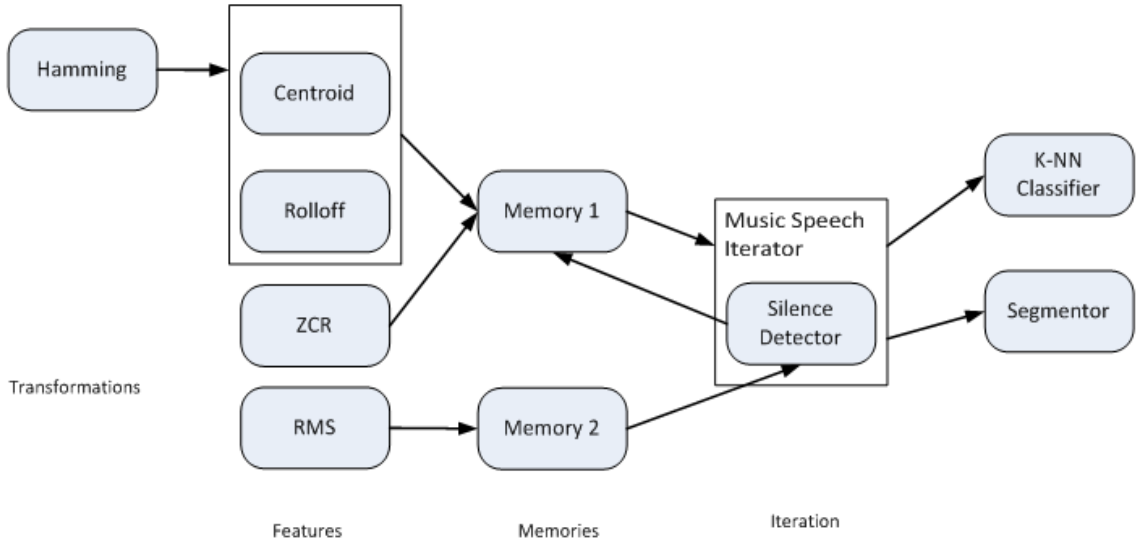


Figure 7-1 MARSYAS system framework

At the beginning, signal power (RMS) is used to discriminate silence/non-silence segments without the need to calculate statistical features. Then, silence segments are excluded from music/speech discrimination to avoid misclassifying them as speech or music. For speech/music discrimination, a more complicated set of nine features rather than energy were used, namely Spectral centroid, Spectral moments, Spectral flux, Pitch, Harmonicity (refers to how strong the sound spectrum; it is calculated by normalizing the frame's spectrum to the perfect line spectrum (Wold et al., 1996)). Mel-Frequency Cestrum Coefficients (MFCC), Linear Prediction Coefficients, RMS, and ZCR. In the machine learning stage, two classifiers are used to categorise, Gaussian (map) and K-nearest neighbour. Each frame is classified with reference to the distance from the training dataset. In other words, the class is determined on the nearest neighbour in the training data set.

One of the significant characteristics of the MARSYAS system is the ability to perform real-time analysis and data processing, and it has been employed in many areas of audio-related work. Some examples are: songs analysis (Yahoo Research), automatic audio segmentation and classification tools (ORBIT Project (Object Re-configurable Broadcast Infrastructure Trial, BBC Research)), music mood, gender classification (male/female/silence), recognition of music emotion, automatically recognizing noise sources like aircraft noise, railway noise, road traffic noise, etc.), automatic subtitle timing annotator (start and end times).

7.1.2 Limitations of MARSYAS

One of the significant limitations of MARSYAS and many mainstream audio information retrieval tools is the lack of ability to handle overlapped audio classes, due to the exclusive classification scheme used. This results in loss of information and inability to tackle mixed elements of the audio content. The decision-making upon a soundtrack with speech over music is either speech or music depending on the intensity of the two, which can result in the loss of potentially useful information. The MARSYAS categorises the audio based on segments of approximately 1second in length into either speech or music, but not overlapped music and speech.

7.1.3 Proposed Algorithm Framework

The proposed system deploys the MARSYAS Framework (Tzanetakis and Cook, 1999b) and associated algorithms, but extends its capability to allow for the kind of overlapped content found in real soundtracks from media archives. Audio content analysis and semantic information retrieval might be viewed as the combination of speech recognition, music information retrieval and event sound detection; each can be achieved using well-developed and dedicated systems or tools. Therefore, the key to

success is a pre-processor that can classify and segment soundtracks, clean the signals, and then feed them to the relevant recognizers. A prototype of this experiment has been implemented using the Matlab compiler and MARSYAS software, which has been written using low-level C code.

The test audio file is segmented into small segments (almost 1 second) to satisfy MARSYAS decision output; also, literature reviews have introduced better accuracy if the feature spaces are integrated over larger time windows.

I Silence Detection

Short time energy reflects the temporal envelope of the signal and is strongly related to the signal content and variation over time. It has been used as a threshold to detect silent frames and intensity changes. Any segment having energy less than statistical threshold value is deemed to be a silent. The threshold value is automatically determined using the histogram technique.

$$E = \frac{1}{L}\sum_{n=1}^{L}\left| f(n)\right|^2 \qquad 7\text{-}1$$

where n and L, are time index, window length value respectively. $f(n)$ refers to the signal frame. The energy for each segment is calculated and if the segments have energy below the determined threshold, they will be assigned as silent; otherwise, the segments will be furthered classified by MARSYAS into either the speech or the music class.

II Redeployment of MARSYAS

Each non-silent frame is further processed by the MARSYAS framework. The problem with exclusive classification and with MARSYAS is that the classification depends on the intensity of the soundtrack contents. Thus, most of the speech over music sound-

tracks may be recognised as music class and this leads to missing out speech information. To handle this problem, MARSYAS is employed twice, before and after applying a speech enhancement algorithm to detect speech utterances and background music at each time. Further cleaning algorithms are applied to the music class segments (more details in the next Section); it is worth noting that combining MARSYAS with one of the speech recognition system through this framework will deliver usable metadata from audio files.

III Overlapped Soundtrack Detection

The system described in this chapter re-deploys the MARSYAS framework that has been tailored by means of adopting additional speech cleaning algorithms, thus avoiding the need to the fully re-implement the classification model. The use of speech enhancement algorithms allows us to maximise the amount of extracted information using logical classification. Also how the power of non-exclusive classification can solve the problem of real-world audio soundtracks with overlap condition has been indicated.

The proposed method as illustrated in Figure 7-2 adopts this approach to non-exclusive problems. A music-cleaning algorithm has been derived in this chapter from an established speech-cleaning algorithm by incorporating zero crossing rates as a discriminating criterion for iterative estimation of the residual (music) spectra, thus spectrum subtraction for speech cleaning has been modified to remove music from mixed segments, which contain speech and music.

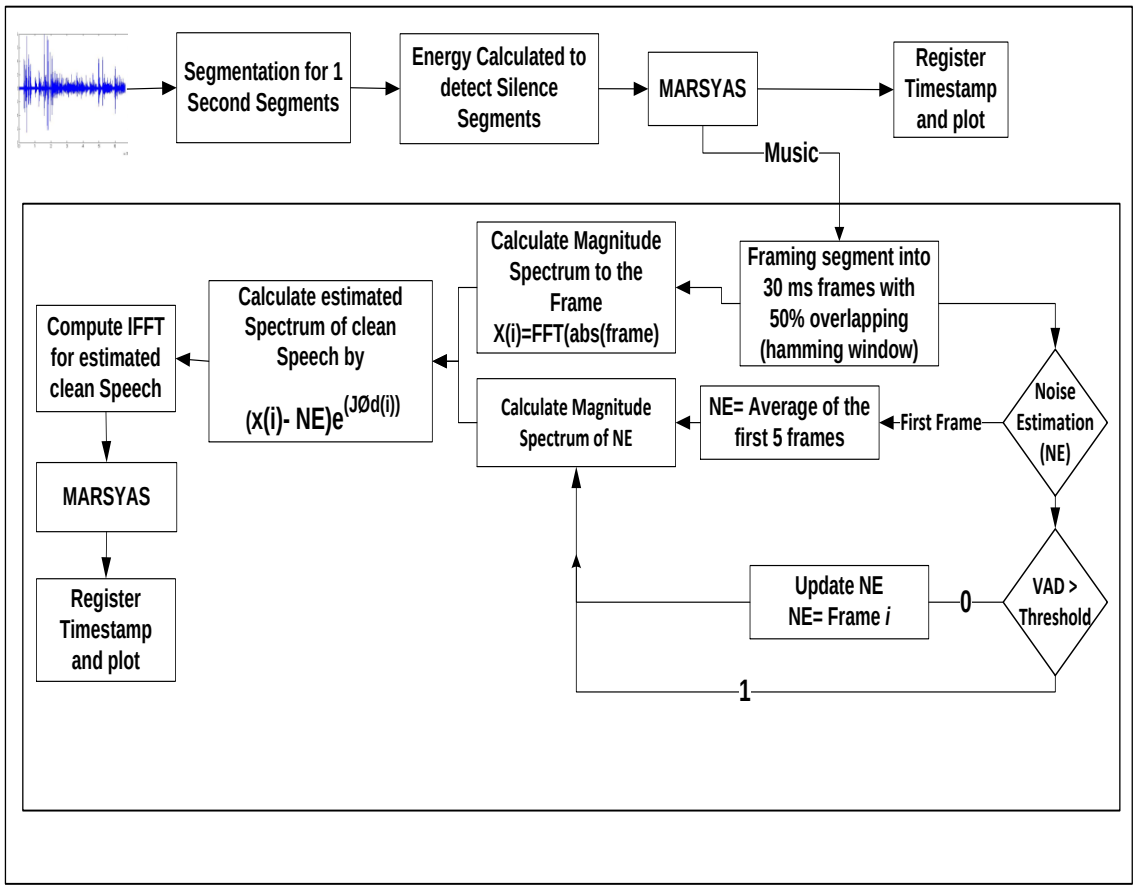


Figure 7-2 MARSYAS with Speech Enhancements Algorithm for Non-exclusive Audio Indexing

Many researchers have focussed their work on speech enhancement or noise reduction. A number of algorithms are proposed to solve this problem and these algorithms have been classified by Kaur1 et al. (2013) into three mainstream techniques; filtering, spectral restoration, and speech model-based parametric methods;

The spectral subtractive approach was introduced for the first time by Weiss as a method for enhancement of the intelligibility of speech which has been corrupted by wideband noise through improvement in the signal-to-noise ratio (Weiss et al., 1975). There are different types of spectral subtraction algorithm has defined in the literature. Upadhyay et al. (2015) carried out a simulation study to compare all these types. The authors concluded that the results of the classical spectral subtraction algorithm mostly results in

audible remnant noise. Many researchers have proposed enhancing the spectral subtraction method through improving the noise estimation. Liu et al. (2013), indicates that the main problem of the conventional spectral subtraction algorithm is that it depends on the stationary noise hypothesis. However, the common energy noise in the real-world soundtracks is non-stationary. Therefore, he proposed a modified spectral subtraction algorithm for estimation of the noise and speech for each frame in the soundtracks using a log spectral distortion. Upadhyay and Karmakar (2012), introduced the enhancement algorithm to reduce the remnant noise, therefore improving the quality of the speech. The proposed method used multi-iterative enhancement, where the output from the first iteration is used as the input to the next iteration. The same method has been introduced to remove musical noise from speech utterances by Miyazaki et al. (2012).

In general, based on the principles of the spectral subtraction algorithm, it has been proposed that the noisy speech signal can be separated into clean speech and noise. Spectral subtractive algorithm is one of the conventional algorithms for speech cleaning from background noise and can be viewed as an adaptive filter that removes the noise according to its spectral content. As the "noise" in this study is music, a modified spectral subtraction has been developed.

The following strategy is proposed and published by Mohammed et al. (2015) for detecting music spectra and then speech enhancement:

- The audio file is segmented into one-second time segments as mentioned above to make deployment of MARSYAS more efficient.
- Segment energy is calculated using Equation 6-1 to recognise silent segments depending on a pre-set threshold for segments which have energy less than the statisti-

cal threshold (this being verified based upon 1000 samples from the benchmark database, which is deployed in this study).

- Voiced segments are analysed with MARSYAS to generate a music timestamp and for further processing, whereas the speech decision is plotted directly to the speech timestamp without the need for further examination.

- For music soundtracks following Loizou (2013), assuming that the musical signal contains clean speech and musical noise based on the spectral subtractive approach concept, see the following Equation 7-2 (Loizou, 2013).

$$s(n) = x(n) + y(n) \tag{7-2}$$

where *s(n)*: Noisy Signal.

x(n): Clean Speech.

y(n): Noise.

- Based on Equation 6-2, the clean speech can be calculated as given in Equation 7-3

$$\hat{x}(n) = s(n) - \hat{y}(n) \tag{7-3}$$

where ^ symbol represents estimated value.

Then, calculate the discrete time FFT for both sides, which could be represented as shown Loizou (2013, p.94) in Equation 7-4

$$\hat{X}(n) = S(n) - \hat{Y}(n) \tag{7-4}$$

- The estimation of musical spectra and noise phase is required to calculate clean speech through the polar form, which is defined in Equation 7-5.

$$\hat{X}(n) = |\hat{X}(n)|\hat{e}^{(j\varphi y(i))} \tag{7-5}$$

where *Øy(i)* represents the noisy speech phase.

The default method of initial musical estimating by the spectral subtractive approach is by assuming that the first five frames of the audio file signal are speech-absent (Loizou, 2013, p. 93). Therefore, to get the music spectrum estimation to improve the speech classification in non-stationary musical noise, the following two steps are carried out. Firstly, MARSYAS is utilised to detect music mood and then select the magnitude spectrum related to that mood, which was trained and calculated previously. Secondly, it is necessary to update spectral estimation from time to time (in each frame) and to preserve the estimation accuracy. Voice Activity Detection (VAD) derived based on ZCR and Signal to Noise Ratio (SNR) in combination are used as threshold values to find music soundtracks and detect music segments which do not contain any speech utterance. Then the average spectrum magnitude is calculated again for these segments to update the estimated value.

- The phase of the music spectra angle is unknown but can be replaced by the phase of the source angle as suggested by Loizou (2013, p. 97). In this experiment, the same assumption is followed to get the best result.

- Subtract the estimated music spectrum magnitude from the magnitude spectrum of the musical speech frame, as defined in Equation 7-6 (Loizou, 2013)

$$\hat{X}(n) = \left[|S(n)| - |\hat{X}(n)| \right] \hat{e}^{(j\varphi y(j))} \qquad 7\text{-}6$$

where *Øy(i)* represents the noisy speech phase.

Finally, the estimated clean time domain signal is calculated by applying the Inverse Fast Fourier Transform (IFFT) on the estimated *X(n)*. Then the timestamp is registered of the speech and music segments. Figure 7-3 illustrates the overall structure of the

framework.

IV Time Stamping

One of the contributions of the proposed architecture is the adoption of the non-exclusive indexing method and one of the primary aims is the timestamp indexing of the audio soundtracks, by recognising that real life soundtracks may have overlaps of speech, music, and event sounds. By preserving all these, and presenting a de-noised version of them to recognition algorithms, information losses are reduced. In Figure 7-3 (a), MARSYAS is deployed without any speech enhancement, whereas Figure 7-3 (b) gives an example of the output of non-exclusive timestamping from the proposed algorithms using Spectral Subtractive Algorithms, MARSYAS and silence segment detection through a MATLAB script. The figure shows the output screen from a 48-second audio file comprising: first, 3-second silence, then 15-second pure music, 15-second pure speech, and followed by a last 15-second mixture of speech and music. The output has detected some of the overlapped occurrences between speech and music classes in addition to speech, music, and silence.

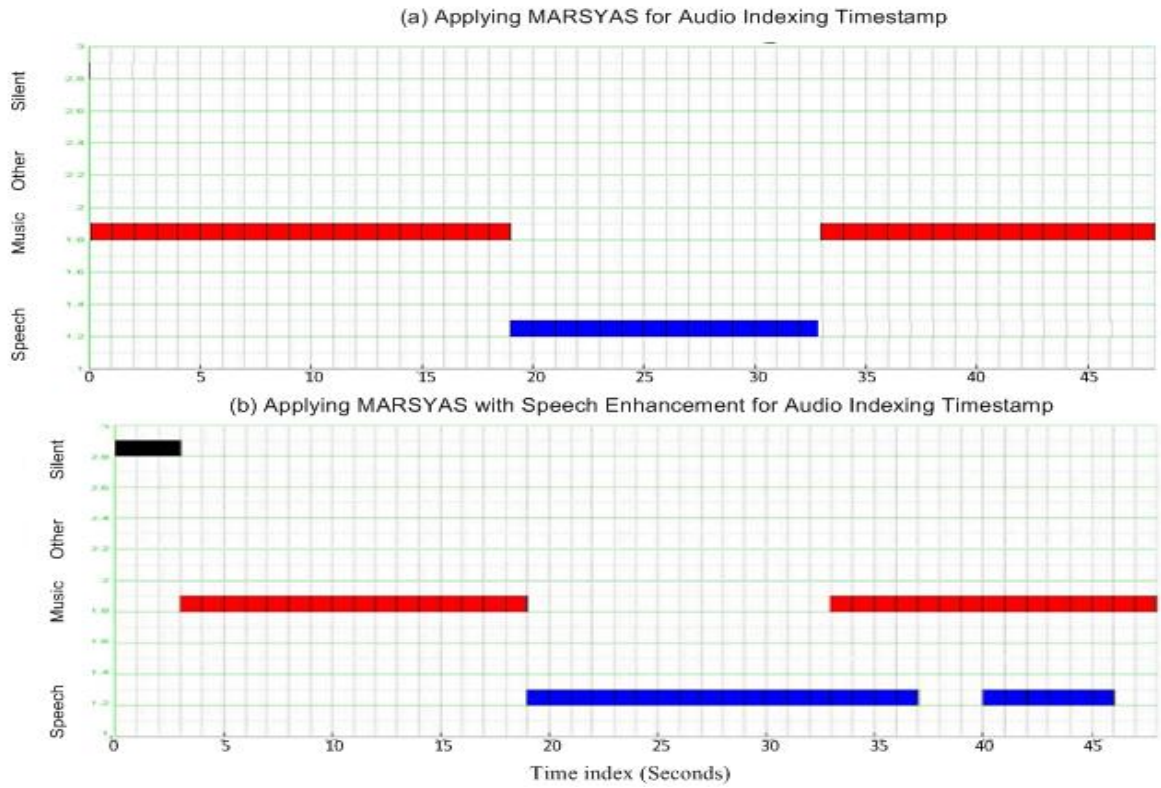


Figure 7-3 The Audio Content Timestamp (top) before the enhancement and (bottom) after the enhancement.

7.1.4 Experimental Setup

I Mixing Stage

In this experiment, 125 speech excerpts with 125 music excerpts, samples from the previously discussed dataset have been mixed in 8 ratios as shown in Table 7-1. Both the music and speech are normalised to the same peak and then loudness level respectively before mixing to ensure that both of them have the same level, so that they will satisfy the required mixing ratio.

Table 7-1 BENCHMARK AUDIO MIXING DATABASE (% of Amplitude)

Speech	30%	40%	.....	80%	90%	100%
Music	70%	60%		20%	10%	0%
Equivalent (Speech-Music Ratio) in dB	-15 dB	10 dB		15 dB	20 dB	Pure Speech

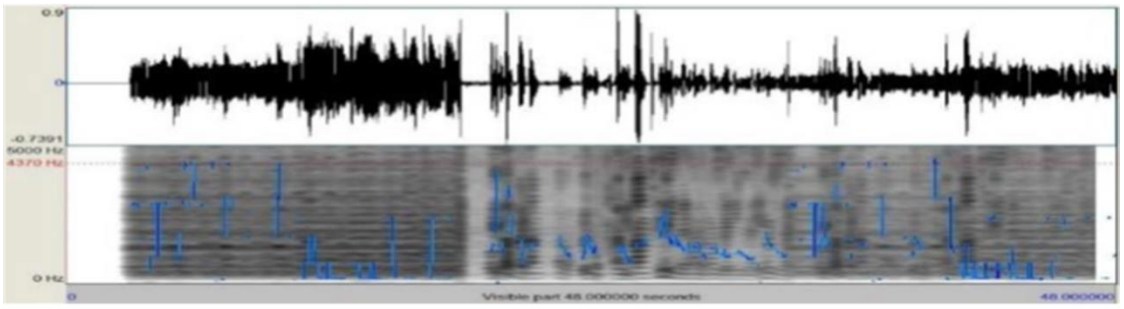
After the speech and music signals were normalised, each of them is multiplied by the predefined number which represents the mixing percentage ratio. Each audio dataset contains 125 speech samples which are mixed with 125 music samples to produce 1000 mixed samples distributed in 8-mixing groups as listed in Table 7-1 according to the speech-music mixing ratio. The reason behind this empirical mixing process is to see how exclusive classification can be made to work with non-exclusive soundtracks and the limitations thereof.

II Training and Validation Results

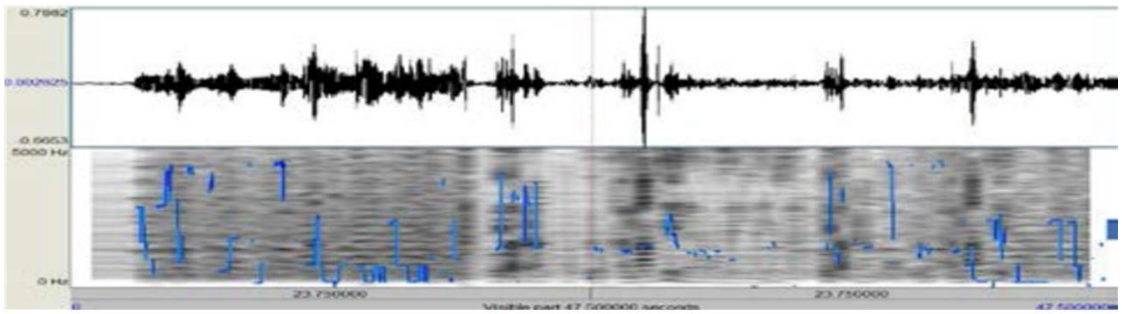
As mentioned previously, MARSYAS is implemented in this core engine because it has the capability for real-time exclusive classifying and by IMIRSEL as an effective evaluation tool for digital music libraries and MIR algorithms. The MARSYAS software was used at the beginning of this study, before reemploying it in the proposed framework. It was used as the evaluation platform for the proposed mixing strategy to create audio benchmarking data due to it having a high accuracy, to compare the speech and music detection accuracy after each of the normalisation stages. In this particular instance, MARSYAS was unable to classify non-exclusive soundtrack data effectively since the classification decision was either speech or music for most instances, and this leads to the loss of useful information. Through this framework, MARSYAS has been redeployed, combined with speech enhancement algorithms to translate an audio file into an audio content timestamp which indicates the start and end of each of audio class (silence, speech, music, and speech over music). This technique has the ability to recognise overlap between speech and music classes. As mentioned previously, in this experiment 1000 mixed speech and music samples at 22050 sampling rate were trained to detect overlapped classes based on feature change intensity. Also, the number of

music mood samples from the MARSYAS website (Tzanetakis, 2015) trained to calculate initial musical noise estimation after music mood is categorised by MARSYAS.

Figure 7-4 shows the spectrogram of the audio file before and after applying speech enhancement; the same file is represented in the timestamp output Figure 7-3. This file was generated from a 47-second audio file comprising the following: - first 2 seconds silence, then 15 seconds pure music, 15 seconds pure speech, and the last 15 seconds are speech over music. The speech and music in the last part are mixed by the same strategy shown in the mixing stage section. It is clearly seen in Figure 7-4 that the cleaning algorithm removes some of the background music, which was very supportive to MARSYAS for categorising speech utterances better in the second iteration even in the presence of slight music background. Praat software was used to plot and analyse audio spectrograms (Paul Boersma, 2011) in the training stage.



a: input source mixed signal spectrogram



b: output of the enhanced signal

Figure 7-4 Audio file spectrogram before and after enhancement

The Table 7-2 shows MARSYAS detection accuracy of speech before and after enhancement in addition to the music. The output of two iterations is combined together

through the timestamp technique as shown in the Figure 7-3. The accuracy ratio denotes the number of frames correctly classified compared to the total number of frames.

The first three mixing groups were not included in this experiment due to the loudness of speech therein being very low compared with the music. As Table 7-2 shows, there is a significant improvement in the speech detection ratio, which is increased by almost 58%.

Table 7-2 Speech Detection Accuracy (SDA) of the Proposed System (before and after enhancement)

Mixing Ratio	SDA before enhancement %	SDA after enhancement %	Music %
S30_M70	11.17	69.61	88.82
S40_M60	13.75	70.93	86.24
S50_M50	16.04	72.10	83.95
S60_M40	17.77	73.23	82.22
S70_M30	18.82	73.37	81.18
S80_M20	20.67	72.01	79.32
S90_M10	23.01	68.96	76.98
S100_M0	85.17	85.53	14.82

7.1.5 Conclusion and Discussion

This experiment has shown that MARSYAS does not work well with overlapped content. Furthermore, it misses out useful information when the overlapping takes place between the audio content. In addition, the experiment showed that the application of spectral subtractive algorithms for speech enhancement could improve MARSYAS classification of overlapping segments (speech over music) through audio content time stamping as shown in Figure 7-3. However, the algorithms need further development to detect and estimate ‘musical noise’ more effectively. This might include updating the

VAD algorithm to work on detecting speech over music using multidimensional feature space. In addition, there is a need for estimation of the music phase because the music does not have stationary characteristics like noise, so this could lead to greater accuracy than when it is replaced by musical speech (speech over music) in Equation 7-3.

The conclusion from this experiment was that classical classification is logically exclusive classification. Hence, instead of the logical classification which assumes that the segment is a member of only one class, since audio segments can be speech, music, event, or any combination of them, there is therefore a need to develop the implementation of non-exclusive audio classification methods to address this problem. In terms of machine learning, empirical experiments have indicated that machine learning for exclusive and non-exclusive classification can then be applied at the same time through tree decision.

7.2 Speech and Music Classification of mixed soundtracks Using Random Forest Evaluation

In this section, an explanation of a baseline classification system using RFs decision tree with short-time feature space will be provided as a 1st method (Meth.1) for overlapped soundtrack classification (baseline classification). Hence, the performance measurements of this experiment will be used in Chapter 10 as a baseline to compare with a suggested high-level framework for mixed soundtrack segmentation in the subsequent chapters of this work.

7.2.1 Audio Features

At the beginning, a set of feature is calculated for each frame on a short timescale, i.e., each signal is framed into a series of consecutive analytical frames with 50 percent overlapping of the window, and for each of these frames a feature value is measured.

The feature extraction stage is represented by a matrix with size $m \times Nof$, where *Nof* refers to the number of extracted features, each single column denoting a particular feature vector. The feature space is calculated for each class (speech, music and mixed). The calculated features and the corresponding adopted window (frame) length are demonstrated in Table 7-3. For more details regarding the calculated features, see Chapter 3.

Table 7-3 Extracted Features and adopted window for each calculated feature

	Feature	**Frame Length**
fe$_1$	ZCR	50 ms
fe $_2$	RMS	50 ms
fe $_3$	Entropy	50 ms
fe $_4$	Brightness	50 ms
fe $_5$	Pitch	50 ms
fe $_6$	Roughness	50 ms
fe $_7$	Irregularity	50 ms
fe $_8$	Spectral Roll off Frequency	50 ms
fe $_9$	Spectral Centroid	50 ms
fe $_{10}$	Spectral Spread	50 ms
fe $_{11}$	Spectral Skewness	50 ms
fe $_{12}$	Spectral Entropy	50 ms
fe $_{13}$- **fe** $_{34}$	MFCC Coefficients (1-22)	50 ms

7.2.2 One Vs All-Classification

Strides have been made toward discrimination between speech and music. The multi-class classification using One Versus All (OVA) or One Versus Rest (OVR) classification techniques has been employed. The objective of such a scheme is training N_c binary classifiers equal to the number of classes *c*, each of them responsible for discriminating between corresponding positive samples against all other negative samples (Rifkin and

Klautau, 2004). In this experiment, the classification problem of overlapping speech and music decomposes into two sub-problems, one for classification of speech against all and the second for classification of music against all. The reason for employing such a technique is to mitigate the classification difficulties in the case of overlapping between speech and music, i.e. this method could be considered for non-exclusive classification through classifying the overlapped segments as speech with the speech classifier and as music with the music classifier. For instance, for the speech training module, the goal is to build a module to identify samples with speech occurrences (pure speech and mixed) against all other classes, (in this example, it is only pure music).

7.2.3 Random Forests Classifier

In the present section, an explanation of the binary RFs classifier that constitutes OVR structures is presented. It is worth noting that CART (Classification and Regression Tree) algorithm by Breiman et al. (1984) is used in this study. Before the clarification of random forests classifiers, which has been applied to classify the respective feature space as speech or music, a concise description of the training phase will be provided, which is considered an essential phase in the classifier's design. Hence, the testing stage can be readily conducted to present the final decision.

Primarily, the constituents of random forest training should be generated and organised before the construction of the classifier module. These constituents are a set of feature vectors of the respective training samples combined with a single target vector that represents the known class labels. For training dataset generation, the features in Table 7-3 have been deployed as a training set as shown in Equation 7-7.

$$v = \{\mathbf{x}_i \left(\mathbf{fe}_1 ... \mathbf{fe}_{34}\right), \mathbf{y}_i\}, i = 1...m \qquad 7\text{-}7$$

The mentioned features have been extracted from the dataset classes (speech, music and

mixed samples) separately. Moreover, in order to generate the respective target vector **y** with size $m \times 1$ for a known training dataset, the samples that contain speech (pure or mixed with music) will be labelled as 1 and the remaining samples (without speech) labelled as 0 in the case of the speech classifier. It is worth noting that these numbers do not represent a restrictive choice, for example, the letters A and B or the numbers +1 and -1 can be used instead of 0 and 1.

To decrease the convergence time of machine learning and increase the learning performance, the following two steps before the training were applied:

- Ensure every feature has been normalised to the mean to decrease the convergence time; this can be done by using Equation 7-8 (Aksoy, 2001)

$$\mathbf{fe}_i = \frac{\mathbf{fe}_i - \overline{\mathbf{fe}}_i}{\arg\max(\mathbf{fe}_i) - \arg\min(\mathbf{fe}_i)} \qquad 7\text{-}8$$

where $\mathbf{fe}_i$ represents a feature vector i and $\overline{\mathbf{fe}}$ is the statistical average of that feature vector.

- Shuffling: the efficiency of the training performance is extremely dependent on the order of the training samples, in that the consecutive frames of the utterance are generally similar. Many researchers have suggested and proposed shuffling the training dataset before the training phase to improve the performance. Therefore, after the combination the features of speech, music and mixed samples, they have been shuffled randomly to increase the performance of random forests model through increasing the randomization probability. A common architecture of the training phase is depicted in Figure 7-5.

On the whole, as the training vectors and respective single class label vectors are created and processed, the classifier can learn. This process is called the training phase and represents the essential concept in the machine learning module representation. At the end, two random forests are trained as mentioned earlier, RF1 and RF2, where RF1 refers to the speech module and RF2 denotes the music classifier, are trained. Consequently, the output of the training phase will be represented by a classifier module, which can be used later on in the testing phase to make the final decision.

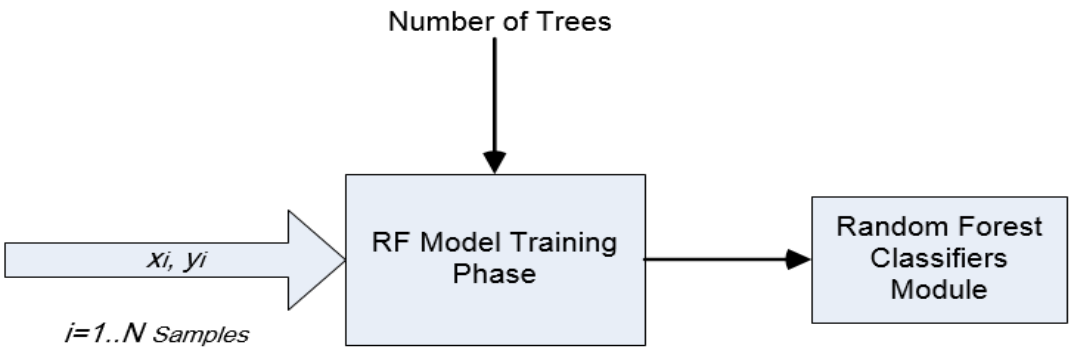


Figure 7-5 General architecture of the training RF phase. The variable xi refers to features

The random forest was trained with different numbers of trees. Thereby 1000 trees size was found to be the optimal size. The reason for this is that the datasets which are used in this study are large in size, and increasing the number of trees above 1000 becomes computationally costly without further significant improvement in performance.

As the RFs classifier's module is trained, it can be saved and downloaded when it is required for testing a new feature space that represents a new dataset. In contrast, in the testing phase, the labels need to be predicted by the classifier's model instead of fed as inputs. The shuffling process does not have any meaning in the testing phase. By contrast, the mean normalisation is an essential step in both the training and the testing phases to present the dataset at the same distribution and scale. For evaluation purposes, to achieve more reliability, any overlap between the training and testing datasets is

avoided, since otherwise the evaluation process will be presented with unfair results. The testing process is presented in Figure 7-6: the feature vectors are extracted from the testing samples, normalised in a similar fashion to the pre-training method, and fed to the pre-trained model. As a result, the output will be represented by a single column vector with the same size as the testing samples $\mathbf{y}=m \times 1$, each element epitomising the predicted label of the respective test sample index, i.e. $\mathbf{y_{10}}$ corresponds to the testing sample with index 10.

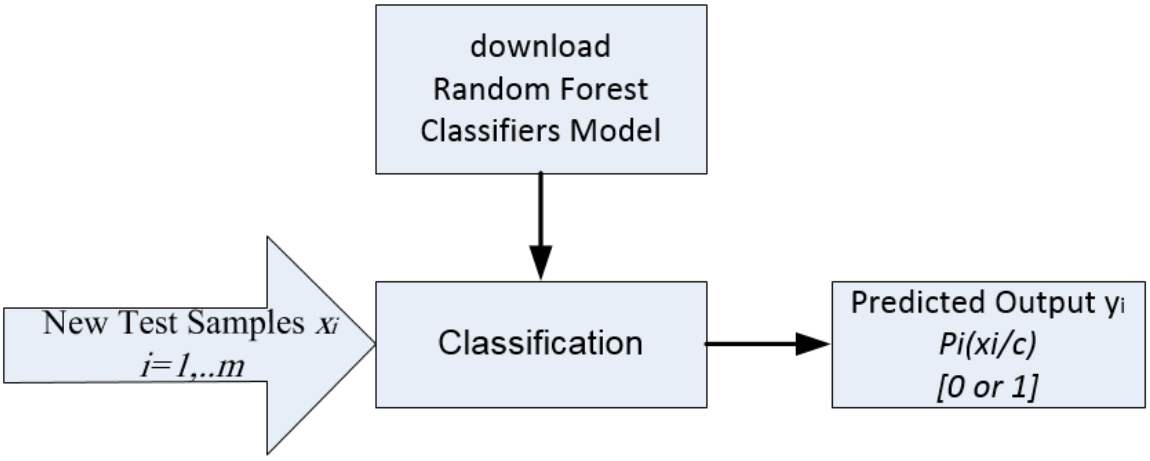


Figure 7-6 Architecture of the testing phase. Feature space is fed as input to classifier's model and the output will represent the predicted class label.

The final decision will be represented by Equation 7-9

$$D = \left\{ \begin{array}{l} 1, if \ \_Sample\_x\_classied\_as(1)by\_only\_RF_1 \\ 0; if \ \_sample\_x\_classified\_as(1)by\_RF_1 and\_RF_2 \\ -1, if \ \_Sample\_x\_classied\_as(1)by\_only\_RF2 \end{array} \right\} \quad 7\text{-}9$$

where 1, 0 and -1 refers to speech, mix and music respectively. The common theme here is the detection of whether the occurrences of the corresponding class are in the given frame or not.

7.3 Summary

In this Chapter two different methods have proposed and developed to solve the problem of the overlapping audio contents classification. The first method represents a pilot investigation into the modification of a well-developed system namely MARSYAS with the aim to improve its capability of handling overlapped content. This has shown to improve classification accuracy in the case of overlapped audio with speech and music from virtual nil to over 60% for speech detection and over 76% for music. However, for typical audio classification without overlap, a performance circa 90% accuracy is generally achievable; therefore, higher accuracy in the overlapped cases is sought after. The study then proceeded to the employment of two binary classifiers simultaneously using the random forests technique, one for speech and the other for music. The target of each classifier is to determine the occurrences of the corresponding class. The results of this investigation show that the accuracy is over 72% when the overlapping takes place between the audio content represented by all overlapped mixing levels while, the classification accuracy represented by the F1 score measurement was over 90% for both speech and music without overlapping. In the next Chapter, the second technique has used with new developed set of features by this study that called augmented features.

8 AUGMENTED AND MODIFIED FEATURES

The post-classification smoothing can improve the classification performance through using a longer frame time-period. In reality, even the human cannot make decisions about audio content based on a very small audio frame.

The suggested features are calculated as statistics for pre-calculated features in Chapter 3 across a number of consecutive frames. Giannakopoulos illustrated six methods for audio segmentation (fixed-window segmentation, probability smoothing, silence removal, signal change detection, speaker diarization, clustering, unsupervised learning, and semi-supervised learning) (Giannakopoulos, 2014, p. 153-180).

A number of augmented features have been tailored to the needs of this study. These have been used to detect speech and music samples in both pure and overlapped cases via naturally recognising the common feature characteristics between pure and overlap. At the beginning, the raw features are extracted; the output will be a two-dimensional matrix. The output is further processed to extract second order statistical features. In this experiment, the frame size is 50 ms and the set of statistical features is calculated based on a band of 20 frames (equal to 1 second). The augmented features can be listed as follows:

8.1 ENTROCY (ENTROpy frequenCY)

Existing features for classification are predominantly established on artificially tailored, non-overlapping audio clips, as illustrated in Chapter 3. A real-world audio soundtrack might include one of the segment types explained earlier. An alternative feature has therefore been proposed for recognising music occurrences; it has shown promising improvements for cases where the music is pure or overlapping with other classes. The

essential aim of applying the frequency calculation as a feature is to indicate the variation in or the distribution of the randomness with respect to a number of consecutive frames. This represents a combination of ENTROpy and frequenCY and therefore it has been christened as *ENTROCY*.

Entropy and Frequency calculations have evolved separately. Entropy was proposed for the first time, as mentioned previously, by Shannon (1948) to measure the amount of information in a signal through calculation of the pdf of each sample in the frame; the randomness distribution of the data is also represented. Entropy is deployed in various classification problems and has hitherto provided adequate results through its ability to detect the complexity of a signal. The range of its application extends across images, automatic speech recognition, health and ecology. In (Toh et al., 2005) the same feature was applied on the sub-band of short-time Fourier transform (STFT) and combined with some of the MFCC bands to improve the results of ASR in a noisy environment; this feature was called Spectral Entropy. MAXENT technique refers to the MAXimum ENTropy model proposed for the first time by Berger et al. (1996) as a statistical model for natural language processing and since applied in many diverse applications. Entropy is also used by Misra et al. (2004) for calculating the entropy of audio spectra for discriminating clean speech/noisy speech, and is proposed as a feature for robust ASR.

8.1.1 EXPERIMENTAL METHOD

The applied classification scheme began by resampling each audio file to a standard sample rate of 22.05 kHz (100 ms). Next, each file was framed, using a frame size of 1102 samples with 50% overlap. This degree of overlap was selected as a trade off with increasing the frequency resolution (number of frequency bins). Then, the Entropy is calculated for each frame. For the entropy calculation, the following procedure was applied: probability of each sample in the frame is calculated following Stewart (2009)

as given in Equation (8-1).

$$Pr_{f(n)}(x_i) = Pr(s \in S \mid f(s) = x_i), i = 1, 2, 3, ..., \quad 8\text{-}1$$

where *S* represents the symbol space of the i^{th} frame, then the sum over the probabilities of all outcomes in the sample space corresponding to the i^{th} frame must be equal to 1.

Let **H** be a vector of entropy features (*1*... *NF*) extracted from *NF* frames; this is calculated using Equation 8-2 (Shannon, 1948).

$$\mathbf{H}_i = -\frac{1}{\log_2(L)}\left[\sum_{n=1}^{L} Pr(f_i(n)) \log_2\left(Pr(f_i(n))\right)\right] \quad 8\text{-}2$$

- Entropy **H** is normalised to the logarithm of the frame length *L* to eliminate the dependency on frame length. Therefore, the entropy value necessarily falls in the interval [0, 1], where the value 1 refers to maximum randomness. The scaling has a significant effect in that it causes the gradient to descend much more quickly and converge in fewer iterations. The experimental results show that most of the music genre frames have higher randomness (entropy) than those of speech.

- The calculated entropy vector **H** is then segmented into segments of size 32 samples, for frequency calculation purposes. The assumption here is 'the classification decision is based on the variation in behaviour across a number of consecutive frames at the same time (visualisation of sound)'. For example, an audio file comprising car engines, babble noise, cars moving, shutting and the opening of bus doors is more likely than not to be a bus station.

- The segmentation is applied with a window moving by one sample at a time on the calculated entropy vector (**H**), if $\mathbf{H} = \{x_1, x_2, \cdots, x_n\}$. The first and second segments will be as shown in Figure 8-1 Entropy Segmentation:

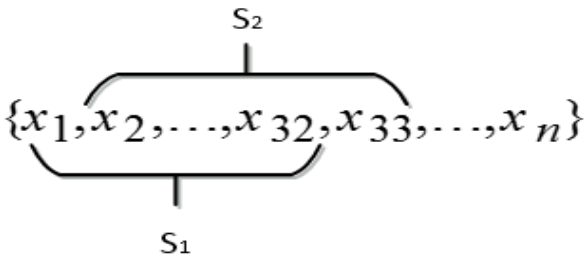


Figure 8-1 Entropy Segmentation

- For spectral analysis purposes, each segment is multiplied first by the Hanning window.
- Applying DCT for each segment, the DCT method is used to calculate the variance of each set of 32 adjacent entropy values.
- Based on the feature importance measurement, which is calculated using RFs, the most important two DCT-coefficients are selected and the rest of the coefficients are omitted.
- The centre of gravity (SC*)*, which reflects the spectral shape of the i^{th} entropy segment, is calculated with reference to Equation 3-14 of the first coefficients part (16-DCT coefficients).
- The average of the segments here will not be equal to zero. Therefore, the first coefficient is ignored in order to eliminate the mean.
- The final output is represented by only three coefficients (3^{rd} and 5^{th} coefficients of DCT plus the frequency's centre of gravity represented by the calculated SC).

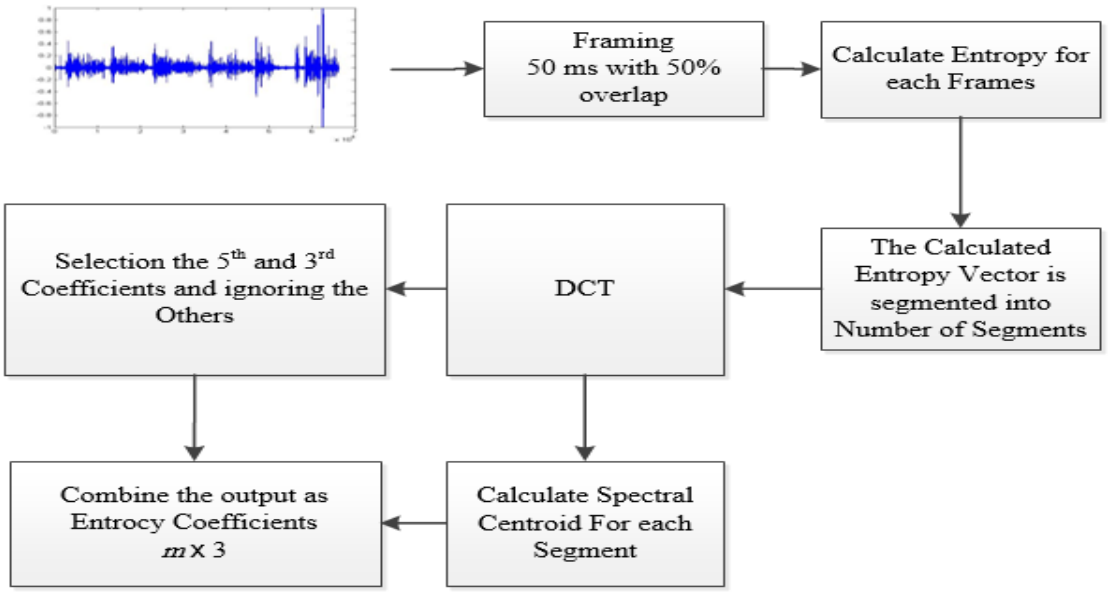


Figure 8-2 Entrocy calculation procedure, m refers to the number of segments

The suggested method is simple and computationally efficient. The Entrocy calculation is done without any computationally expensive optimisations or sophisticated mathematical operations performed in any of the calculation stages described above. The method is straightforward to understand and relates to audio content, complexity, and homogeneity. Moreover, the construction is general and can be applied to most audio information retrieval studies such as music/speech discrimination, segmentation, MIR, or classification.

8.1.2 Entrocy Validation

For the purposes of Entrocy validation, an experiment to build a module for detection of music against all other classes has been carried out; the samples that contain music score (music or mix) will be labelled as 1 and the remaining samples will be labelled as 0. The machine learning method used is the Random Forests technique (see Chapter 4) trained with varying numbers of trees; it has been established empirically that 1000 trees are an optimal size as said before. Figure 8-3 shows a simple random forest, ex-

plained in the subsequent chapter, with 20 trees using the Entrocy feature. Class 1 represents music, class 2 represents other classes without music occurring; the features number represents the Entrocy coefficients, and the numerical value represents the threshold of hyperplanes with respect to that feature, which is determined by the Random Forest technique.

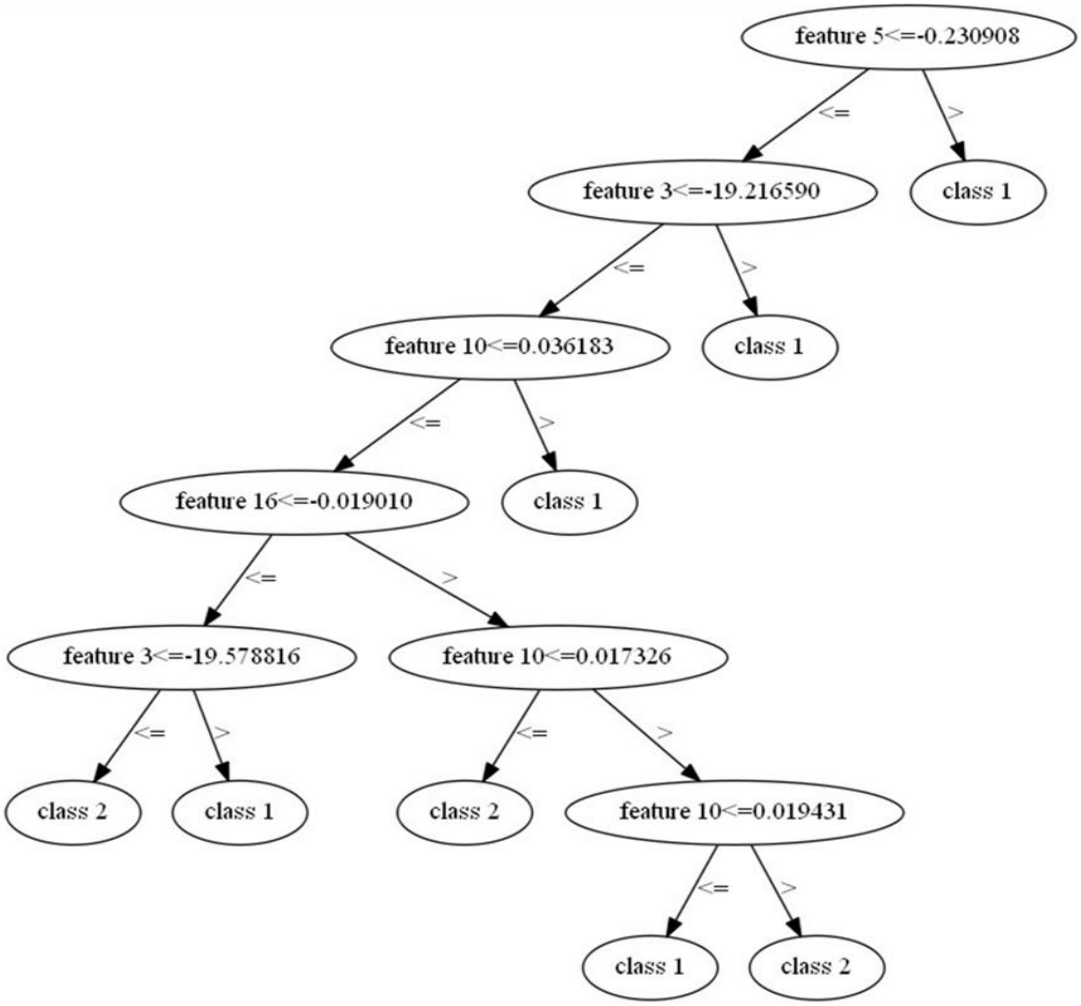


Figure 8-3 Random forest DT, the value represents the threshold of hyperplanes with respect to the feature axis

The basic concept of random forest trees is that they work as a collection of hyperplanes (thresholds), where each threshold is orthogonal to the corresponding feature axis (Kargupta et al., 2006). Figure 8-4 shows the visualisation of the 2-Dimensional features (Entrocy coefficients 5 and 3) where the fifth coefficient is localised as the X-axis and

the third coefficient as the Y-axis with their corresponding hyperplane thresholds.

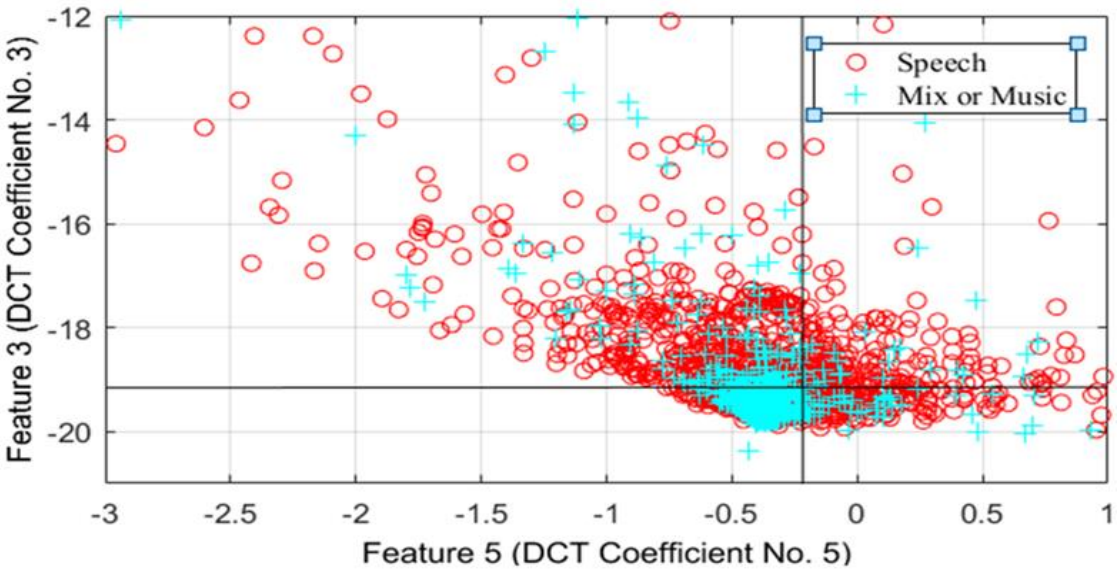


Figure 8-4 2D Feature Spaces with their respective thresholds from Figure 8-3

8.1.3 Entrocy Results and Discussion

The results of Entrocy have been compared with the common speech/music discriminator features (namely ZCR, Spectral Centroid, Spectral Entropy, Roll-Off and Chroma). The most significant results for pure music / pure speech discrimination were where ZCR or Entrocy was used with all samples selected from the GTZAN dataset described in Section 5-1. Table 8-1 and Table 8-2 illustrate the results of the two classification scenarios. The error rate is defined as the number of misclassified samples as a fraction of the total number of samples. The music group included all samples with music backgrounds, as pointed out previously (M, SM, SME and ME), and the group without music comprised all other samples (S, SE and Event). From Table 7-1, it can be seen that the results show that ZCR outperformed other features since speech samples have a higher rate of zero crossing than pure music. However, it failed when used to discriminate between the with and without music classes due to an under-fitting problem that led to

non-convergence. By contrast, Entrocy manifests a high level of performance in music/non-music discrimination.

Table 8-1: Speech/Music discrimination Error Rate, the ratio between number of misclassified frames and total number of frames n each group

Features	Pure Music	Pure Speech
ZCR	10.33%	9.75%
Entrocy	13.60%	17.48%
Spectral Entropy	49.53%	63.85%
Spectral Centroid	60.84%	42.13%
Roll Off	42.37%	36.00%
Chroma	29.12%	59.18%

Table 8-2: Music detection Error Rate, the ratio between number of misclassified frames and total number of frames in each group

Features	without Music (pure Speech)	with Music
ZCR	25.87%	82.25%
Entrocy	21.26%	17.46%
Spectral Entropy	34.07%	38.48%
Spectral Centroid	41.40%	44.46%
Roll Off	62.83%	36.78%
Chroma	61.31%	37.62%

Consequently, ZCR has been combined with Entrocy to distinguish samples with music from samples without music. It has achieved the best results when detecting the presence or absence of music with high accuracy. The classification performance of Music, SM, ME, SME, Speech, SE and Event came up to 91.19%,90.03%, 91.31%, 76.68%, 86.82%, 84.34%, and 92.90% respectively.

Finally, pink noise was mixed with speech and music in two different signal-to-noise ratios (SNR) - 20 and 25 dB - in place of an event. Figure 8-5 shows the error rate corresponding to each mixing group. As noted, with this mixing dataset the results presented here may facilitate improvements in the detection of music using Entrocy due to feature learning with ad hoc characteristics or applications. (The results hold when training the system on a specific class of things, e.g. noise here is better than using a wide range of events)

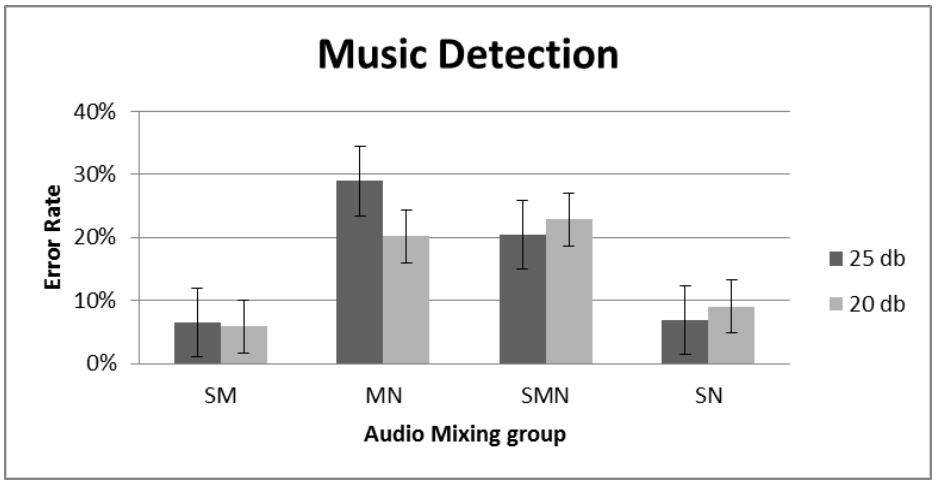


Figure 8-5: Entrocy Error Rate for music detection (S refers to Speech, M: Music and N represents Noise)

A novel feature, 'Entrocy', for classification of audio with overlapping of the three main classes (speech, music, and event) has been proposed in this study. This feature reflects the level of randomness of oscillation of the respective classes over time. It is logical to expect to be able to estimate the homogeneity changes through estimation of the frequency of the segments contents level. This is the key to information preservation, through detecting a particular class even when it overlaps with other classes. This will be a basis for non-exclusive segmentation, which is highlighted by some researchers.

Music, speech, and event sounds can be consistently detected and segmented through building one module belonging to each class. Event sound recognition is challenging,

as there is no prior knowledge of how many different types of events there are likely to be and which are of interest in this application. Therefore, an ad hoc approach might be best for event processing based on the application due to the set of event sounds being an open set. When the Office Live dataset or pink noise is deployed, the Entrocy reflects significant improvements in terms of detection of music occurrences.

It is sensible to conclude that Entrocy provides a cost and time gain for the process of classification due to achieving these results with features of only a few dimensions. It has also been observed that for detection of speech, music, and events, conveniently a longer frame period might be beneficial; in this experiment 50 ms has been used, and the final discernment was with respect to almost 1 second. This is not surprising; even the human listener cannot differentiate sounds on the basis of very short isolated samples.

The experimental results show that the frames that involve music, even when overlapped with other audio classes, have generally lower Entrocy than those with music absent.

8.2 Mean Crossing Ratio (MCR)

The predominant context has a significant effect on harmonic stability (Bharucha and Krumhansl, 1983). This means that a signal with abrupt changes (i.e. generally speech) has lower stability. To measure the fundamental frequency, the method of counting zero crossing is used in speech processing applications (cf. ZCR 3.3.1). In the present study, a method similar to ZCR is applied as a modified feature to estimate the harmonic stability level over a medium timeframe, through estimating pre-calculated short time feature value crossings of its mean per unit time. This means the calculation of the number of times that said features cross over their mean axis per unit time; hence, this is called

the Mean Crossing Ratio (MCR). Firstly, as in the previously mentioned procedure of feature extraction, each 1-second window is framed into 20 frames. Firstly, all the foregoing short-term features are computed for each frame. Then MCR is calculated for successive intervals of 20 values of each extracted feature. To sum up, each MCR value will denote the crossings in a 1-second window time. The MCR for the segment i^{th} is calculated according to Equation 8-3.

$$MCR_i = \frac{1}{2}\left(\frac{\sum_{n=2}^{S} \left| Sign\left(\mathbf{fe}_i(n)\right) - Sign\left(\mathbf{fe}_i(n-1)\right) \right|}{S} \right) \qquad 8\text{-}3$$

where S is the length of the segment and **fe** is the pre-calculated short-time feature.

$$Sign(x) = \begin{Bmatrix} 1 & if\left[x > \mu(f_i)\right] \\ 0 & if\left[x = \mu(f_i)\right] \\ -1 & if\left[x < \mu(f_i)\right] \end{Bmatrix} \qquad 8\text{-}4$$

The hypothesis here is (H_0: μ(MCR) for speech, μ(MCR) for mix>μ (MCR) for music). To prove or reject this hypothesis, the MCR is calculated for all extracted short-term features, explained previously in Chapter 3, over twenty feature values. The speech and mixed group size (Number of samples =13171); the mean value of the speech and the mixed group is in general bigger than the music group for most of the features as shown in Appendix A Table I. The music samples numbered 13145, and to prove the proposed hypothesis an independent sample t-test was performed. As can be seen in Appendix A Table I and Table II, speech and mix were mostly associated with a statistically significant larger mean than music.

The hypothesis is rejected only for the MCRs that were computed on the basis of features: MFCC7, MFCC11, MFCC13, Roughness, Regularity and Centroid. As Figure 8-6 (A) illustrates, the MCR values that correspond to speech or mix are generally higher than those for without speech. Generally, the with speech group has lower harmonic stability and higher numerical values, as indicated in Figure 8-6 (B) by the statistical features curve. This observation is based on a set of 50 randomly selected MCR values for each class, and it can be conceptualised physically as follows: feature values with speech presence much more often change between higher and lower power than samples without speech, i.e. the features encompass changes such as the wave transitioning from low to high power states.

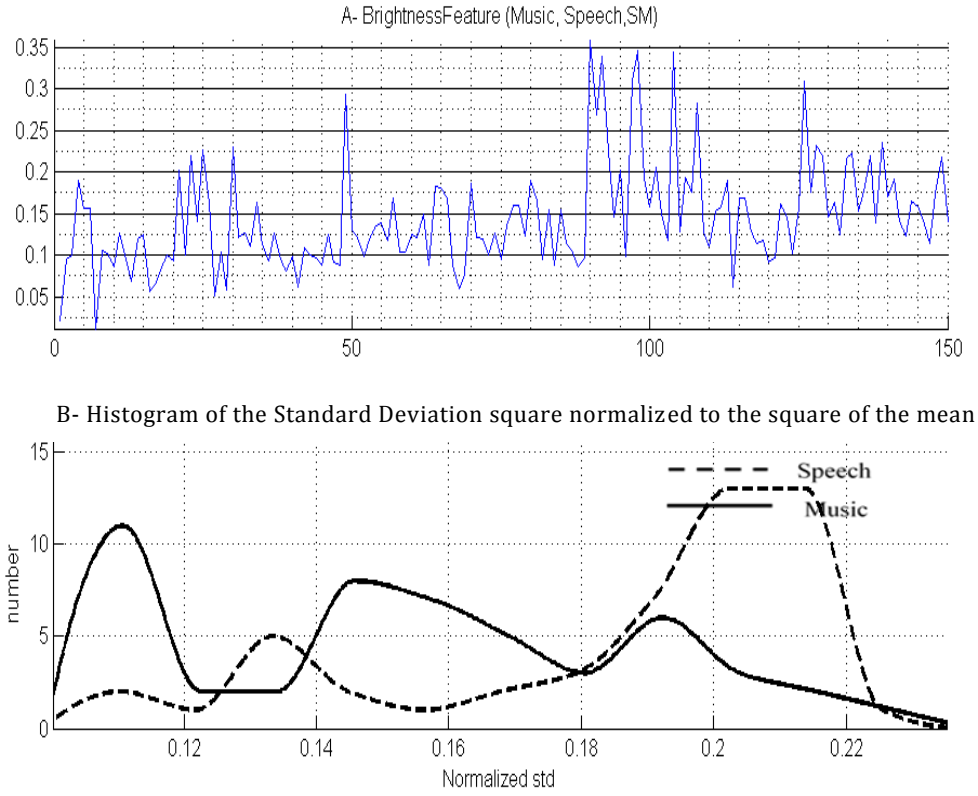


Figure 8-6 MCR statistical features for the Brightness short-term feature, A- Feature Amplitude, B- distribution of with speech and with music segments for MCR values

8.3 Peak Variance Rate (PVR)

One of the most obvious problems in feature learning is a large dynamic range of features. This property leads to poor performance and the under-fitting problem. The features that have such problems do so as a result of using different normalisation schemes. To mitigate the impact of such effects in this work, the peaks variance is calculated. The PVR is computed for each 20 successive intervals for each extracted feature with a window moving by one sample at a time, as shown in Figure 8-7.

At the beginning, local maxima and minima are computed for each 20 successive intervals for all short-term extracted feature with a window moving by one sample at a time. Next, the statistical average of the distance between all peaks (PVR) is calculated according to the following Equation 8-5

$$PVR = \frac{\sum_{n=2}^{j} \left| Pe(n) - Pe(n-1) \right|}{j-1} \qquad 8\text{-}5$$

where j here represents the number of detected local and global maxima for each pre-calculated feature segment. Figure 8-7 presents the general calculation method of *PVR* statistical features.

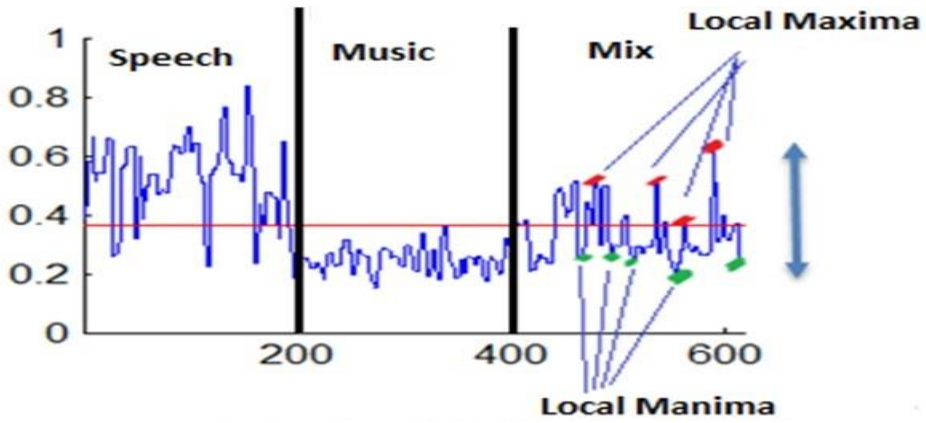


Figure 8-7 Calculation Method of PVR Feature

This proposed technique is deployed for speech presence recognition through detection of potential segments containing higher dynamic range such as in Figure 8-8. A large dynamic range is detected in segment i if the next and preceding segments are significantly separate and distant. In the case of features corresponding to speech samples, they have higher PVR than segments without speech presence, because the distance of variance range is large in comparison with the mean value of music peaks variance. The reason why the distribution of with music in Figure 8-8 (B) appears wider than the with speech group is that the representation of pure music has low amplitude. This is clear in the first part of Figure 8-8 (A), which is related to pure music, while music over speech (SM) segments have higher values because they contain occurrences of speech.

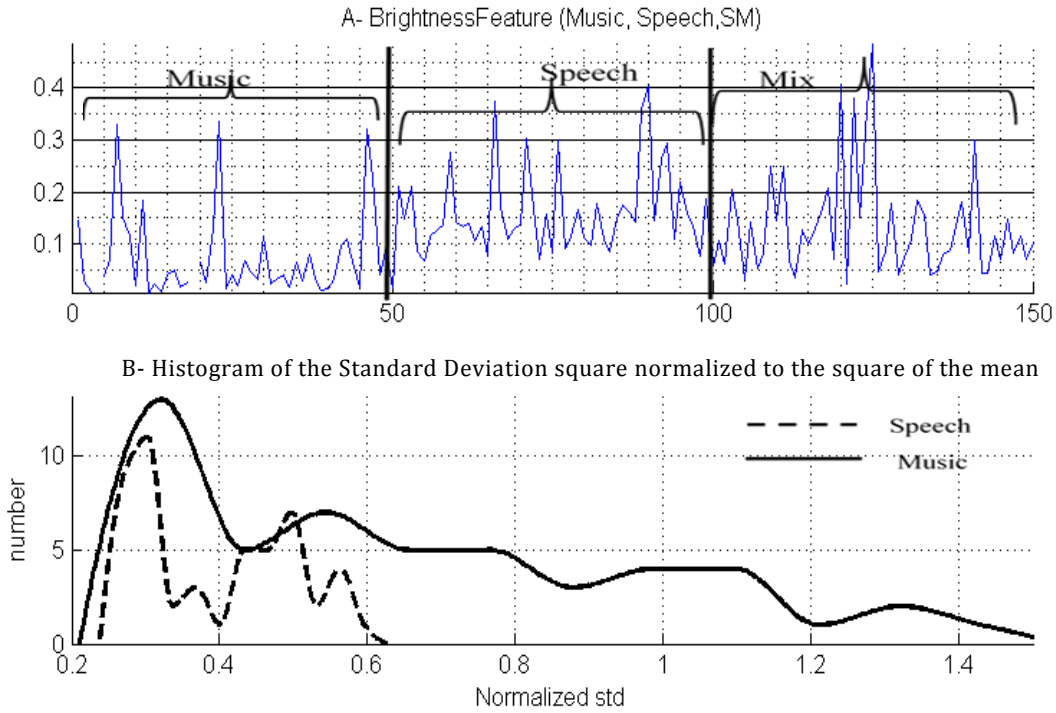


Figure 8-8 PVR Feature, A- PVR Feature Amplitude B- Distribution Histogram of speech and music segments for PVR values

8.4 Speech and Music Classification Using the Augmented Features

The classification system using RFs decision tree with feature space explained in this chapter has been trained and tested as a 2nd method (Meth.2). The trained system therefore has a similar design to that of method Meth.1 in Section 6.2, with regards to applying a different set of features. The performance measurements of this experiment will be represented and used for comparisons in Chapter 10. Figure 8-9 shows the procedure that is followed for mixed soundtrack classification with the augmented features

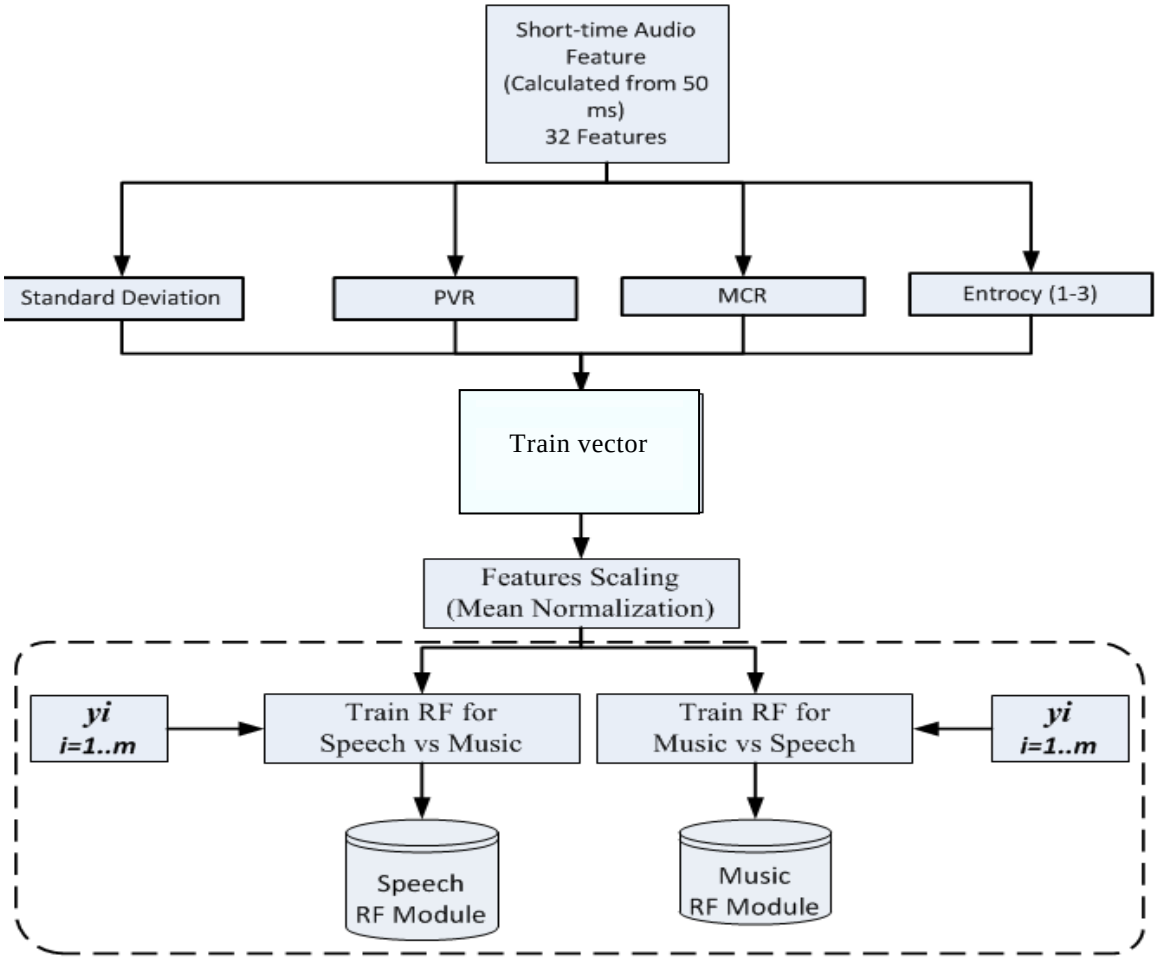


Figure 8-9 Training RF using Augmented Features Space

8.5 Summary

The research in this chapter investigated the use of augmented and medium term features with random forests to tackle the overlapping problem. Three augmented features have been proposed namely Entrocy, Mean Crossing Ratio and Peak Variance Rate. The RFs therefore has trained using the augmented features as shown in Figure 8-6. From the results that will demonstrate in Figure 10-5, it is apparent that there is slight improvement in the detection of overlapped contents to over 75% using the augmented features. The next Chapter will illustrate the designed method using SSA and RFs to identify the contents of soundtracks even in the presence of overlapping for speech, music, or a combination.

9 SSA PROPOSED METHODS: TRAINING AND VALIDATION-BASED SYSTEM

9.1 Introduction

The purpose of this chapter is to explain the methods being proposed, and give details regarding the suggested methods for the classification of overlapped soundtracks using SSA and RFs. Two different experiments were conducted to investigate the ability of SSA to decompose mixed soundtracks into different components with mostly non-overlapping content. A number of audio features were extracted. To this end, random forests decision trees were employed to classify the enhanced output signals from SSA into silence, speech, music, or a combination thereof. An alternative method was also suggested and applied. Instead of reconstructing the filtered components into the time domain and then classifying them, the features that extracted from PCs were fed directly into RFs to classify them as speech or non-speech. Finally, the probability of PCs being classified as speech was used as a voting criterion to classify the corresponding frame into speech, music or a mix.

9.2 Adapted SSA Method Description

Vautard and Ghil (1989) and Elsner and Tsonis (2013, p. 62) have showed that the SSA technique for time series decomposition could be used for spectral decomposition, as a time series generally has varying frequencies but similar amplitude. Consequently, the suggestion here is that SSA can be used to decompose speech and music in mixed soundtracks into a number of almost non-overlapping components, since the mixed soundtracks have similar characteristics of varying frequency.

The present method (Meth.3) applies SSA to the clustering of the speech and music oscillation patterns included within mixed soundtracks into a number of spaces denoted by the elementary matrices, which were clarified in the preceding chapter. The grouping criterion was applied to split these matrices into two clusters, speech and music.

The matrices within each cluster are added together to generate one matrix that reflects significant oscillation, e.g. music with light speech in the background, or vice versa. Next, the final matrices, each corresponding to a different group, are recovered into the time domain. Thus, each of the mixed soundtracks will be represented by more than one time series. This comes before the extracted set of features from each output signal. Next, random forests decision tree classifiers are used to test the "enhanced" output signals to classify them into speech or music. To examine the technique of using SSA to determine significant oscillations, speech and music soundtracks have been mixed together in different speech-music ratios using the technique illustrated in Section 5-2, which has been published previously (Mohammed et al., 2015). One of the mixed sample with time length N_t = 25 seconds is depicted in Figure 9-1.

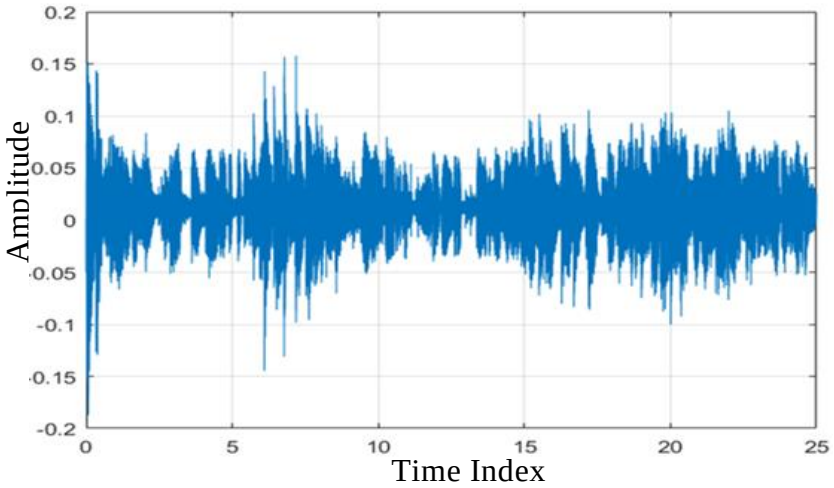


Figure 9-1 Mixed Soundtrack Signal

A flowchart illustrating the procedure for the enhancement of mixed soundtrack components is shown in Figure 9-2.

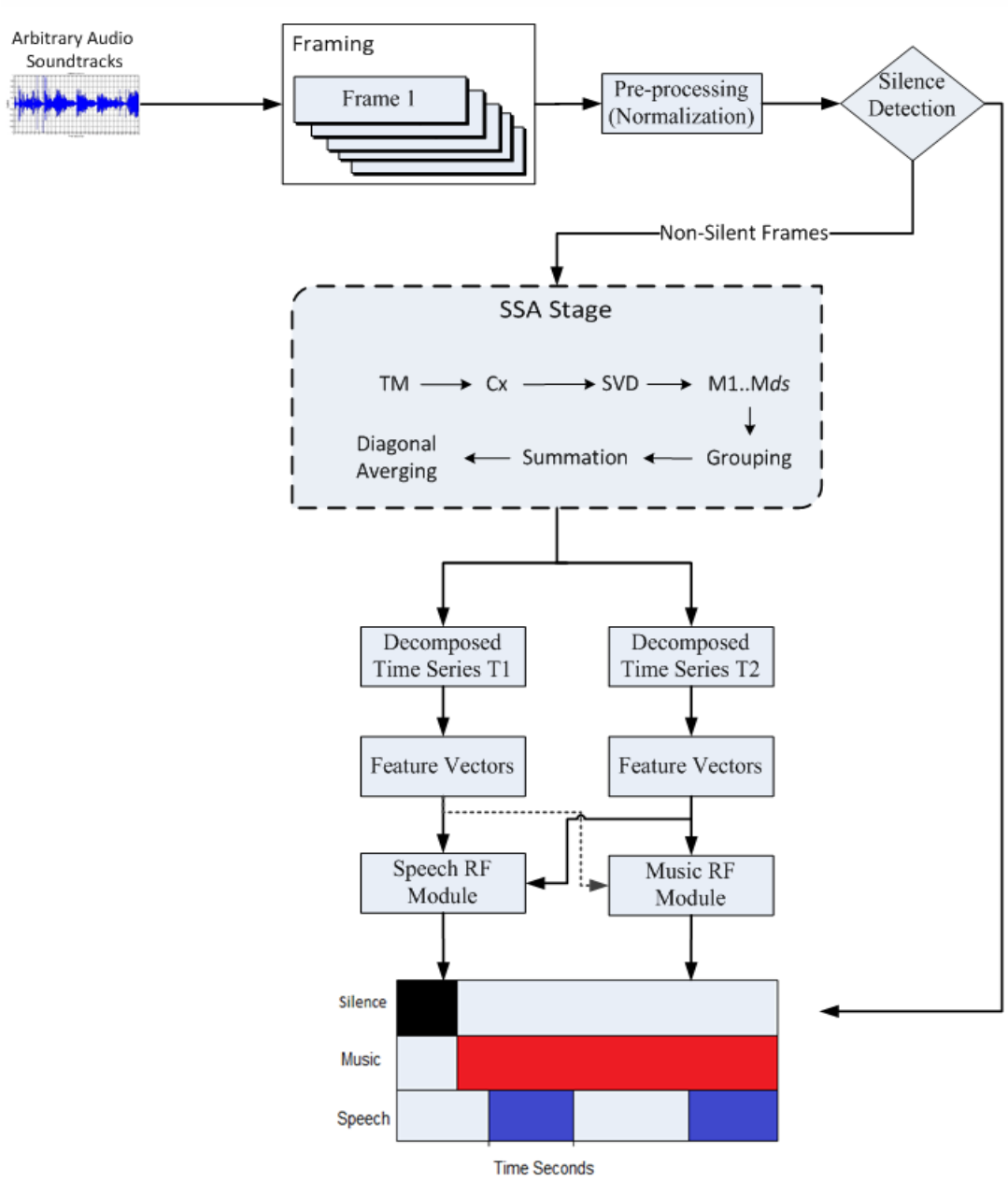


Figure 9-2 General flowchart of proposed method (Meth.3) for mixed soundtrack decomposition.

9.2.1 Window Length Optimization

In order to map the soundtrack vector onto the trajectory matrix, the window length selection must be optimised. As mentioned in Chapter 8, the suitable window length is highly dependent on the aims of the analysis. Therefore, the assumptions surrounding

window length that have been reported by other researchers and considered in the determination of optimal length in this study are presented here:

- Many researchers have determined that the optimal length is $N_t/2$, since the lagged covariance matrix will acquire the characteristics of the symmetric matrix.
- Elsner and Tsonis (2013, p.57) concluded that longer window length could lead the output to reflect the high-frequency components. By contrast, the level of competition between high and low frequencies is lower with a smaller window length, leading to increased statistical confidence.
- However, some researchers have argued that the results of SSA are not significantly affected by window length, as long as it is shorter than the time series under test (Penland et al., 1991). The finding of this study contradicts this somewhat, as will be explained later in this chapter.
- Vautard et al. tested different types of noise with 100 generated time series. They observed the impact of the window length in the determined dominant signal. The authors suggested the optimal window length is $(Fr/5, 2\delta Fr)$, where Fr is the frequency and $2\delta Fr$ denotes the bandwidth. Furthermore, they indicated that a greater window length enhances the detection of the eigenvalue pairs (Vautard et al., 1992). Their recommendation, however, is based on trend forecasting of weather time series, where the frequency of the time series under test is very low compared to that of the audio data. Therefore, it is not possible to apply it to the audio dataset.
- It also has been asserted by many authors that each pair of eigenvalues has nearly similar values with reference to dominant frequency and significant oscillations in the signal (Vautard and Ghil, 1989, Mohammadi et al., 2016, Enshaeifar et al., 2016, Elsner and Tsonis, 2013).

Since there is no specific benchmark length, and the SSA was not previously used for speech/music enhancement, the heuristic method could be used to determine the optimal range, highlighting the smaller variance between the lower subspaces of the eigenvalue set. Accordingly, the optimal window length is the one that enhances the detection of the eigenvalue pairs.

In this study, therefore to find optimal window length, a comparison of the λs of the lagged covariance matrices was made for eight window lengths used to generate the trajectory matrix from the frame with length 480 samples. These window lengths were (8, 48, 80, 160, 240, 320, 400, 464 samples). Figure 9-3 depicts the average of the eigenvalues $\boldsymbol{\lambda}$ measured with the 54142 sets of mixed (speech with music background) time series for the eight aforementioned window lengths. All calculated averages have been normalised to the mean using Equation 9-1 to make the comparison clearer.

$$\hat{\lambda}_{li} = \frac{\lambda_{li} - \overline{\lambda_l}}{Arg\max(\lambda_l) - Arg\min(\lambda_l)} \qquad 9\text{-}1$$

where l=1..., 8 and i represents the length of λ_l; the eigenvalues will be between 0 and 1 (1 in the case of maximum eigenvalue).

The lower subspace eigenvalues (largest 25 λ) are presented in Figure 9-3. In general, concordant with the observations of others, the first two λ are appreciably greater than any other eigenvalues. However, this experiment determined that the optimal window length range that highlights pairs of eigenvalues that have nearly similar values to satisfy the last assumption is 160-320. Moreover, these window lengths demonstrate that there are three pairs of nearly equal values, each corresponding to an important frequency. In contrast, the other selected window lengths showed a deterioration in the ability to determine the pairs of eigenvalues. In addition, this deterioration increases as

the length increases or decreases outside the aforementioned optimal window lengths.

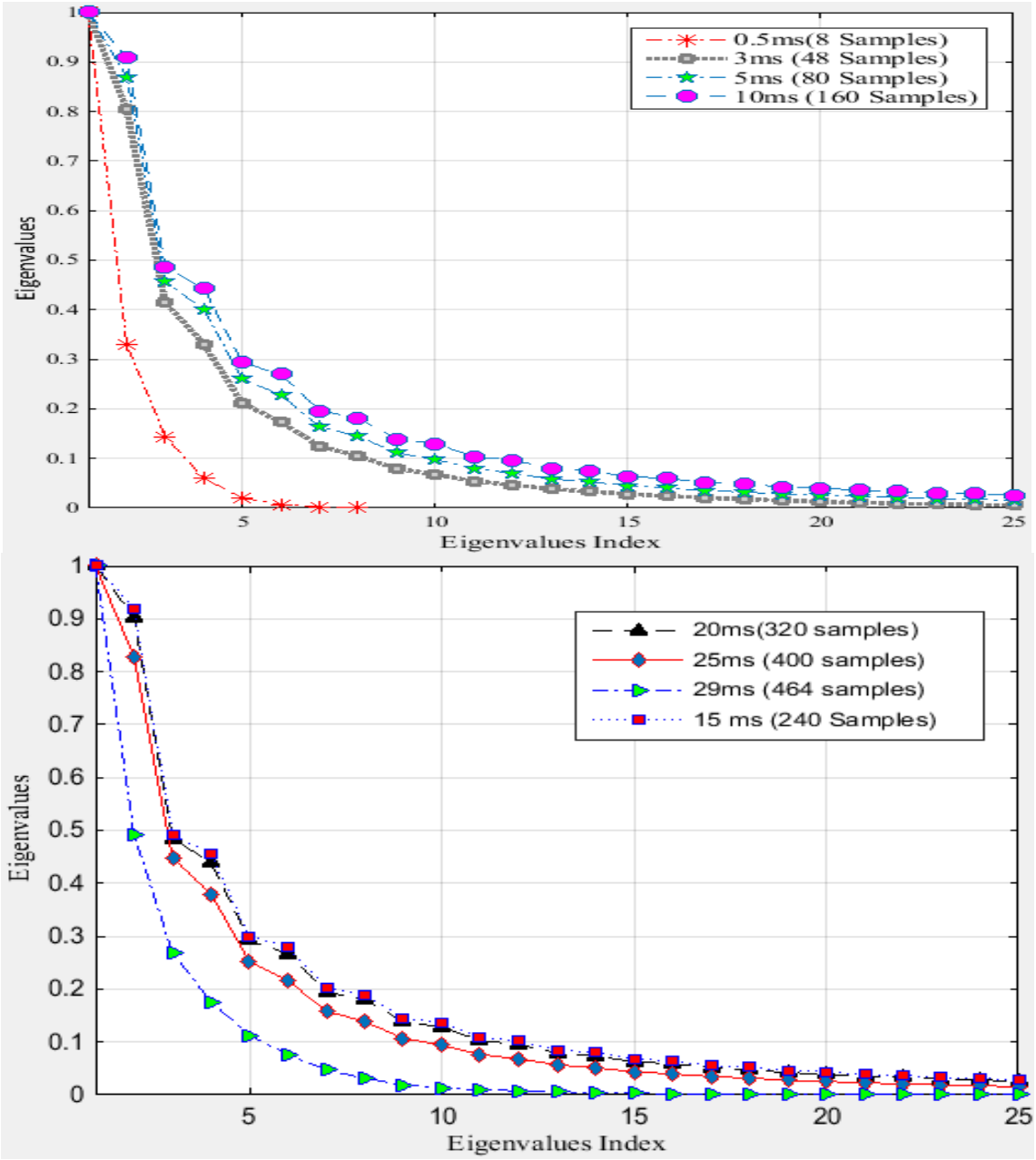


Figure 9-3 Singular eigenvalues of the mixed sample time series for various window lengths, 8-160 samples (top) and 240- 464 samples (bottom).

Consequently, this study recommends that window length (L_w) can be determined using Equation 9-2.

$$\frac{N_t}{4} \leq L_w \leq \frac{3N_t}{4} \qquad 9\text{-}2$$

where N_t reflects the length of the time domain under test. The physical significance of

this size is that it maintains most of the variance of the time series by generating a symmetric or quasi-symmetric matrix.

In the case that L_w is small, this might cause a number of adjacent eigenvalues (variance peaks) in the spectrum of a given time series to merge together and be represented by only one eigenvalue. By contrast, a large L_w value (high resolution) ought to split the variance peaks (eigenvalues) into several consecutive frequency components.

The consensus is that the optimal size is $N_t/2$ because, as most other authors have noted, the trajectory matrix will be represented by a symmetrical matrix. Nevertheless, using a slightly smaller window length, as indicated in Equation 9-2, will achieve approximately the same results. In addition, a shorter length will economise the performance time of SSA; this is considered to be one of the challenges of applying SSA to a huge dataset such as ours, because it consists of an immense number of matrix multiplications, the number of which increases with dataset size. In this example, L_w=160 would be a reasonable size.

9.2.2 Singular Value Decomposition

The eigenvectors belonging to PC pairs of nearly equal singular value were then used to reconstruct the PCs of the time series using Equation 8-10 (each column vector of the PC matrix denotes a separate PC). It is obvious from Figure 9-4, which illustrates the first three PC pairs of mixed soundtrack depicted in Figure 9-1, that each pair represents a specific oscillation. The elementary matrices are then obtained by multiplying each of PC by the corresponding eigenvector and dividing by the square root of each eigenelements.

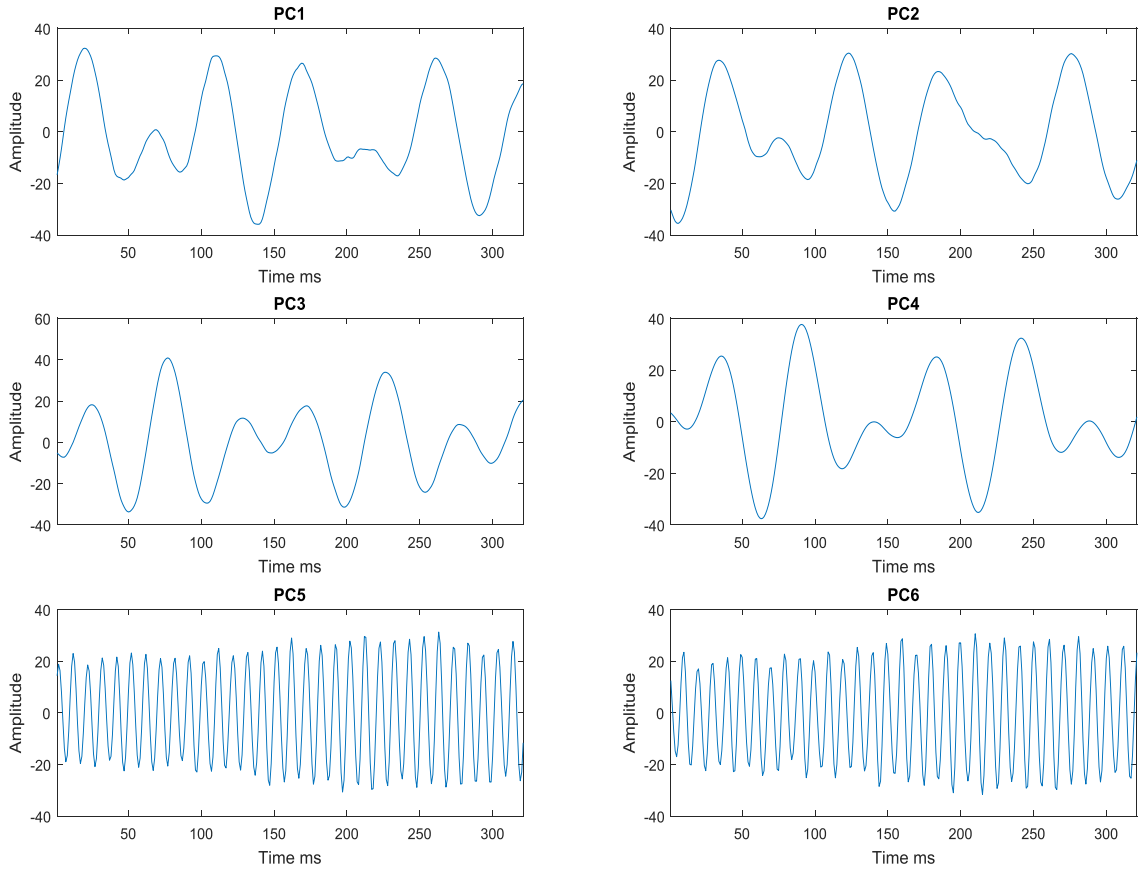


Figure 9-4 First three pairs of PC plotted as time series

9.2.3 Grouping Speech/Music Components

In order to select subsets of eigenelements and their respective Principal Components (PCs) that identify the music and speech components, this research first compared the statistical average of the (PCs) spectrum, which has been extracted from many thousands of frames for the three different audio classes (speech, music, and mixed). The spectrum was measured by computing the DFT function of the statistical average of the principal components extracted from the aforementioned three classes. At the end, the magnitude of the spectrum of all three classes was calculated and plotted. As illustrated in Figure 9-5, all three classes manifest greater amplitude at low frequencies (0-1000 Hz) and lesser amplitude at higher frequencies. The amplitude of the frequencies pertaining to pure speech is higher in the low-frequency range than for music and mixed. By contrast, the music and mixed classes reflect a much wider range of frequencies (0-

4000 Hz) (higher variance) than speech (mostly 0-1000 Hz). This would seem to support the idea that SSA could be used to differentiate varying frequencies with similar amplitude from mixed signals into enhanced speech and music.

Furthermore, a particularly significant finding of the present investigation, as depicted by the results in Figure 9-5, is that speech has a slightly higher auto-correlation than music (i.e. it has higher associated frequencies than the music pattern). Accordingly, the speech variance will be represented by higher space eigenvalues than music, projected onto the higher subspace, which represents lower variances in the frequencies.

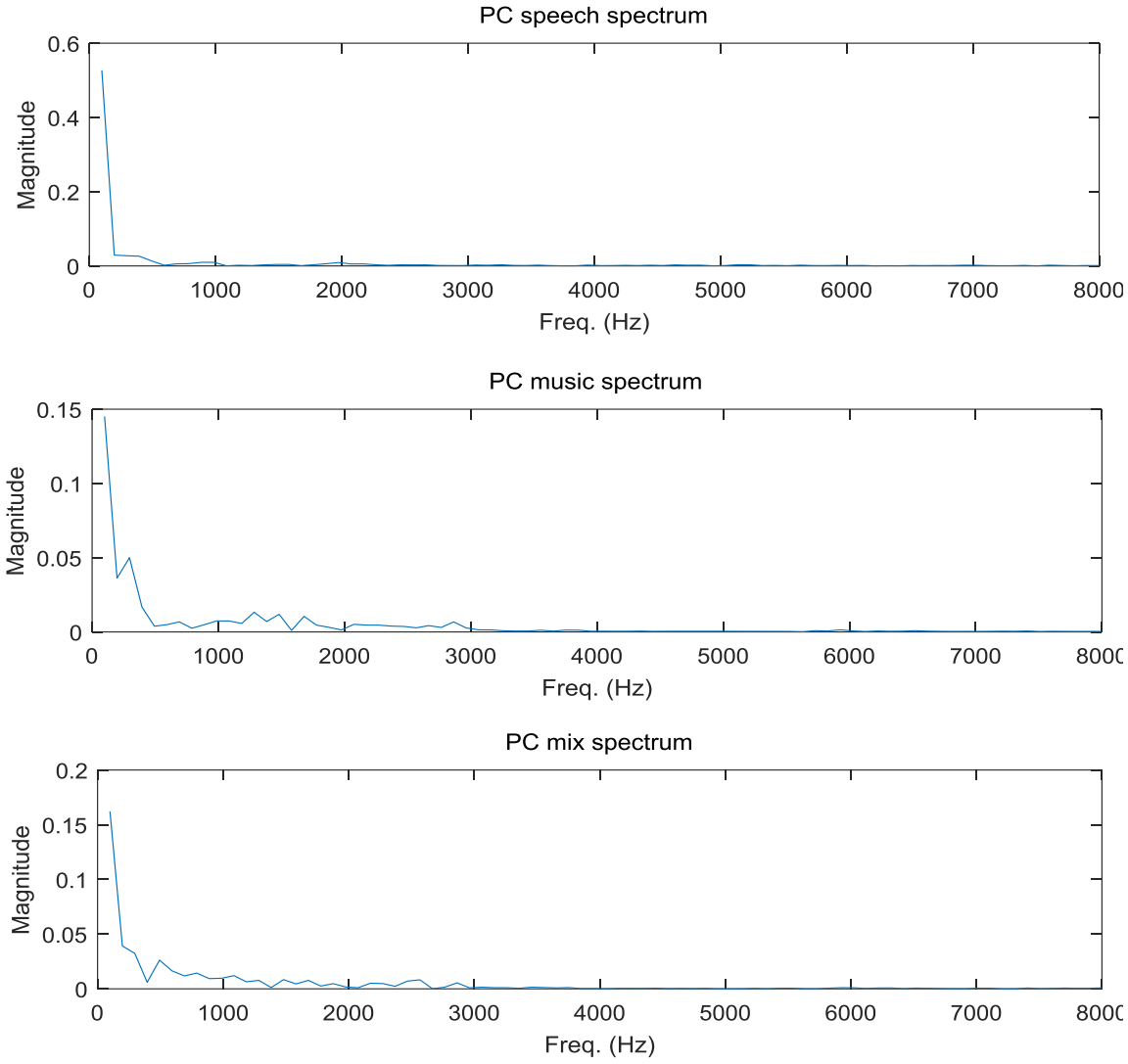


Figure 9-5 The estimated spectrum for PCs of speech (top), the estimated spectrum for PCs of music (middle) and the estimated spectrum for PCs of mixed class (bottom)

In contrast, the music pattern contains higher variances in the frequencies that are projected on the lower subspace of the eigenvalue distribution (c.f. 8.3.4). In view of this finding, the given audio signal will be represented by a number of different oscillations, each reflecting a dominant pattern and other inferior components, as shown in Equation 9-3.

$$I_{input} = I_{music} + I_{speech} \qquad 9\text{-}3$$

where $I_{input}, I_{music}, I_{speech}$ denote the elementary matrix group related to the input signal, the elementary matrices corresponding to music, and those corresponding to speech components, respectively; each of these groups is represented by a different number of elementary matrices. In order to categorise the oscillations using SSA, a proper grouping criterion is required. In this work, *i* and *i+1* is selected as a pair if it satisfies the condition shown in Equation 9-4:

$$\left| 1 - \frac{\lambda_{i+1}}{\lambda_i} \right| \leq th \qquad 9\text{-}4$$

The threshold *th* is selected based on the normalised eigenvalue set, which is plotted in Figure 9-3, and can be changed with regard to the amplitude of the signal and the selected window length. As a result, a particular threshold *th* is set for each PC cluster. Music is usually represented by a higher amplitude than speech. Bearing in mind what was explained earlier, therefore, and since it manifests itself in the lower subspace, a higher threshold *th* value is selected for music (0.1). On the other hand, for speech PCs, which are characterised by lower amplitude and higher subspace, a lower *th* is chosen to be (0.05) (These values are verified based on the dataset used).

Another method for determine the grouping criterion is the weighted-correlation (w-correlation) method. It is a well-known measure of deviation of two series from *w*-

orthogonality. This method has been used for determine the noise boundary by Hassani (2010) and Golyandina (2001) and many other authors. Golyandina (2001) has stated "the Eigen-triples entering the same group can correspond to highly correlated components of the series". The correlations measure is defined by absolute values from 0 to 1, if the w-correlation value is close to 0, then the two series are nearly w-orthogonal (highly separable), but if the value is skew toward 1, then the series are far from being w-orthogonal and (badly separable). Hence, the elementary matrices arriving the same group should reflect highly correlated components of the series. W-correlations matrix for the junior 100 elementary matrices depicted in Figure 9-6. As illustrated the correlations matrix is presented in 21-colour scale from white to black, these colours are related to the absolute values of correlations from 0 to 1. Depends on the figure data, the first and second pair of Eigen-triples can be used for the reconstruction of the filtered series and omit the rest matrices (noise components). The value of w-correlation between the first pair of reconstructed component and the residual is equal to 0.004. Also, It is worth to noting that the first (1, 2) and second (2, 3) reconstructed series were hearable and this supportive by $|\rho^{(w)}| = 0.32$. To achieve the w-correlation investigation, the Caterpillar software (Golyandina et al., 2016) was used.

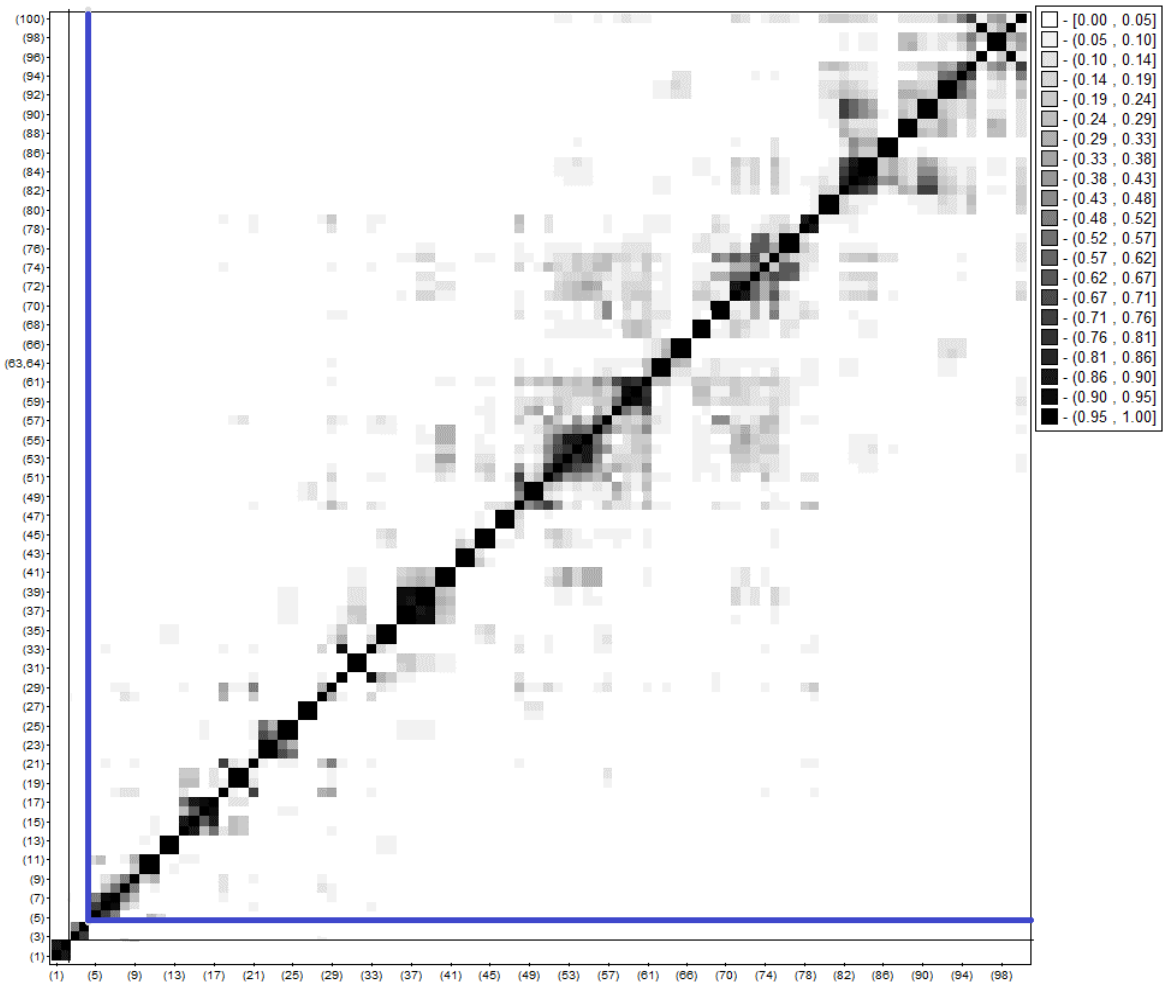


Figure 9-6 matrix of w-correlations, of mixed soundtrack

To conclude, after grouping each pair of resultant elementary matrices together by summation and potentially generating two groups I_1 (I_{music}) and I_2 (I_{speech}) , these could be restored to the time domain by diagonal averaging as described in the preceding chapter (see Section 8.3.4). Figure 9-7 illustrates the reconstructed signal of the first two dominant pairs, combined with the original signal for comparison.

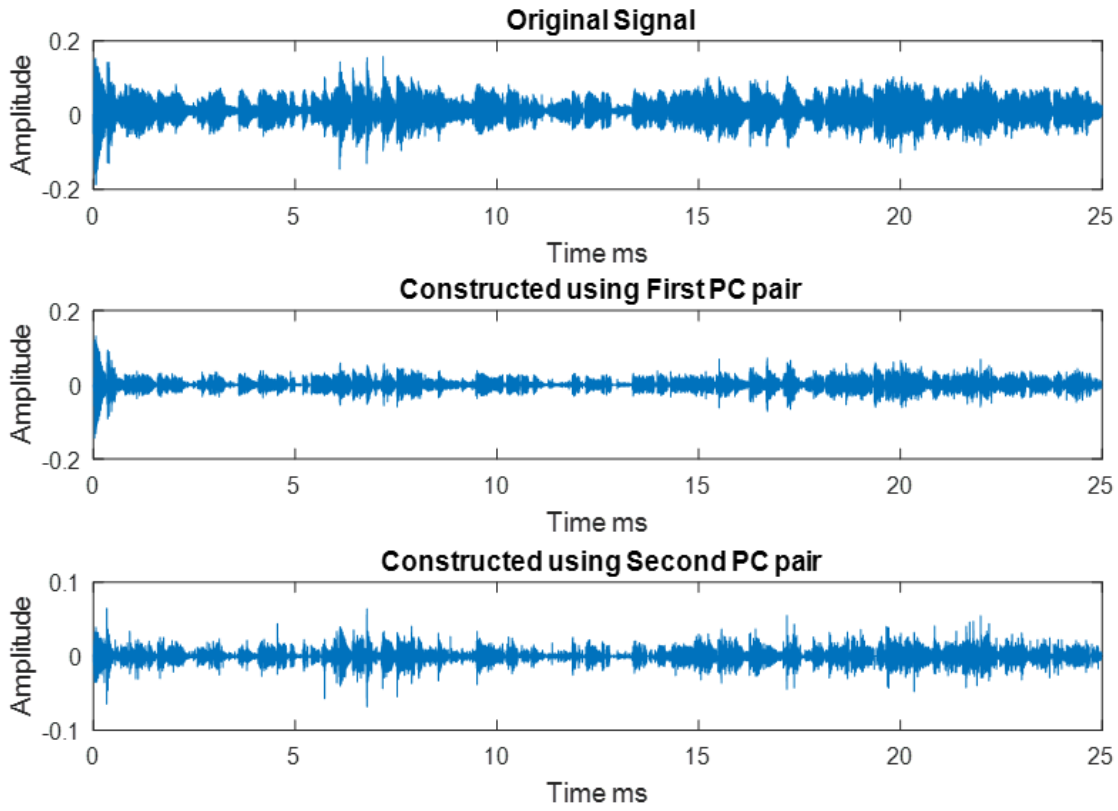


Figure 9-7 Input signal of SSA (top), constructed time signal corresponding to the first pair (middle signal) constructed time signal corresponding to the second pair (bottom)

A comparison between the DFT of the original signal and of the enhanced outputs is illustrated in Figure 9-8. It is clear that the original signal spectrum has greater variance in frequency than the first constructed time series, which in turn has greater variance than the second constructed time series.

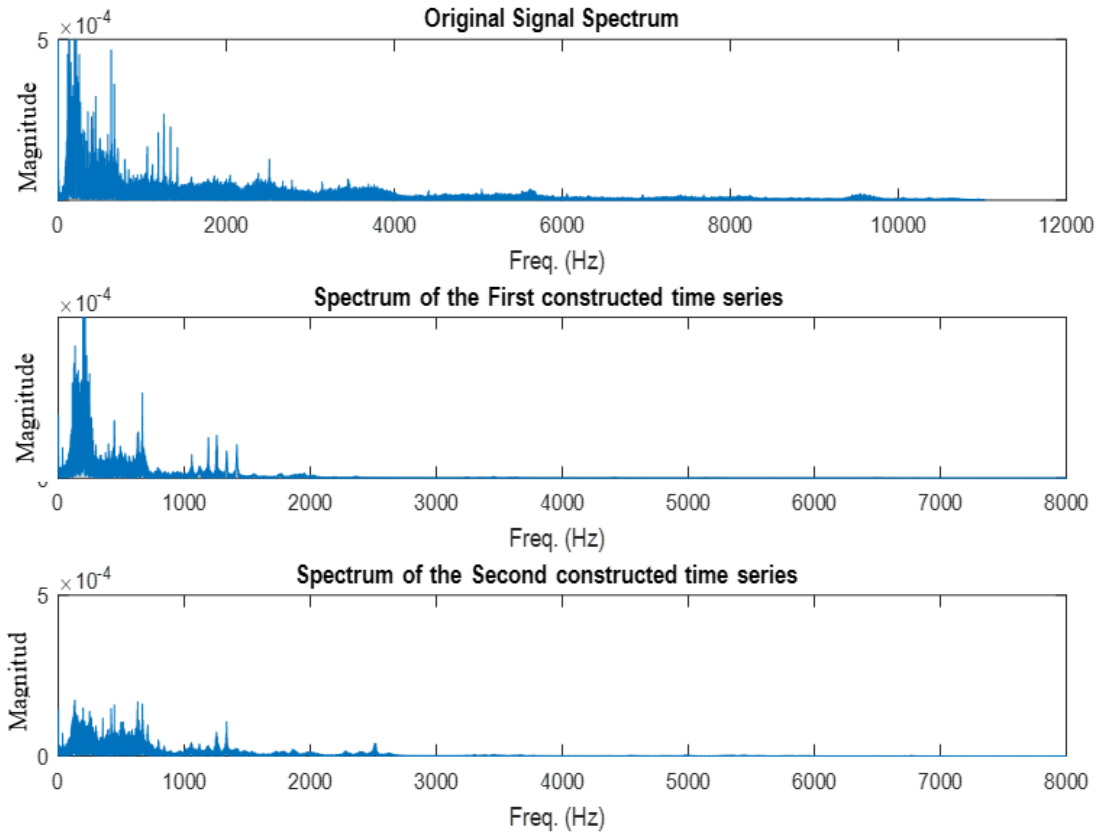


Figure 9-8 Spectrum of the original signal of SSA (top), spectrum of the reconstructed time signal corresponding to the first pair (middle), reconstructed from the second pair (bottom)

9.2.4 Reconstructed Signal Classification

Finally, each of the enhanced pairs of signals from the SSA stage is processed in parallel by machine learning. At the beginning, 34 features are calculated for each reconstructed frame on a short timescale, i.e., each signal is framed into a series of consecutive analytical frames with 50 percent overlapping of the window, and for each of these frames a feature value is measured. The feature extraction stage is represented by a matrix with size $m \times 34$, each single column denoting a particular feature vector. The calculated features and the corresponding adopted window (frame) length are demonstrated in Table 6-3 ($\mathbf{fe}_1$- $\mathbf{fe}_{34}$). For more details regarding the calculated features, see Chapter 3.

In particular, each frame will be decomposed into two-time series $T_y = \{T_1, T_2\}$ using SSA. Thus, the feature space **x** is calculated for each T_y. At the end, two $\mathrm{RF}_z = \{\mathrm{RF}_1, \mathrm{RF}_2\}$ where RF_1 refers to speech and RF_2 to music classifier, are trained for the two decomposed time series. The final decision will be represented in matrix *D*, which can be determined according to Equation 9-5.

$$D = \left\{ \begin{array}{l} 1, if\ Sample\quad x\quad classied\quad as(1)\quad by\quad only\ RF1 \\ 0; if\ sample\quad x\quad classified\quad as(1)\quad by\quad RF_1\quad and\ RF_2 \\ -1, if\ Sample\quad x\quad classied\quad as(1)\quad by\quad only\quad RF2 \end{array} \right\} \qquad 9\text{-}5$$

where 1, 0 and -1 refers to speech, mix and music respectively, the common theme here is the detection of whether the occurrences of the corresponding class are in the given frame or not. Notably, the given frame is decomposed into two-time subseries, and each of these decomposition time subseries is classified separately. For more information about the training and testing phase, see Section 6.2.

9.3 Principal Components Classification Method

An alternative method is proposed as 4th method (Math.4) to classify a mixed soundtrack based on the decomposed PCs for each frame. The results of preceding method guided its development as follows:

- We conclude along with others (for instance Vautard et al., 1992, Sanei and Hassani, 2016) that each PC pair vector corresponding to nearly equal eigenvalue has a unique oscillation pattern, as shown in Figure 9-9, which depicts the calculated frequency of the first 10 principal components.

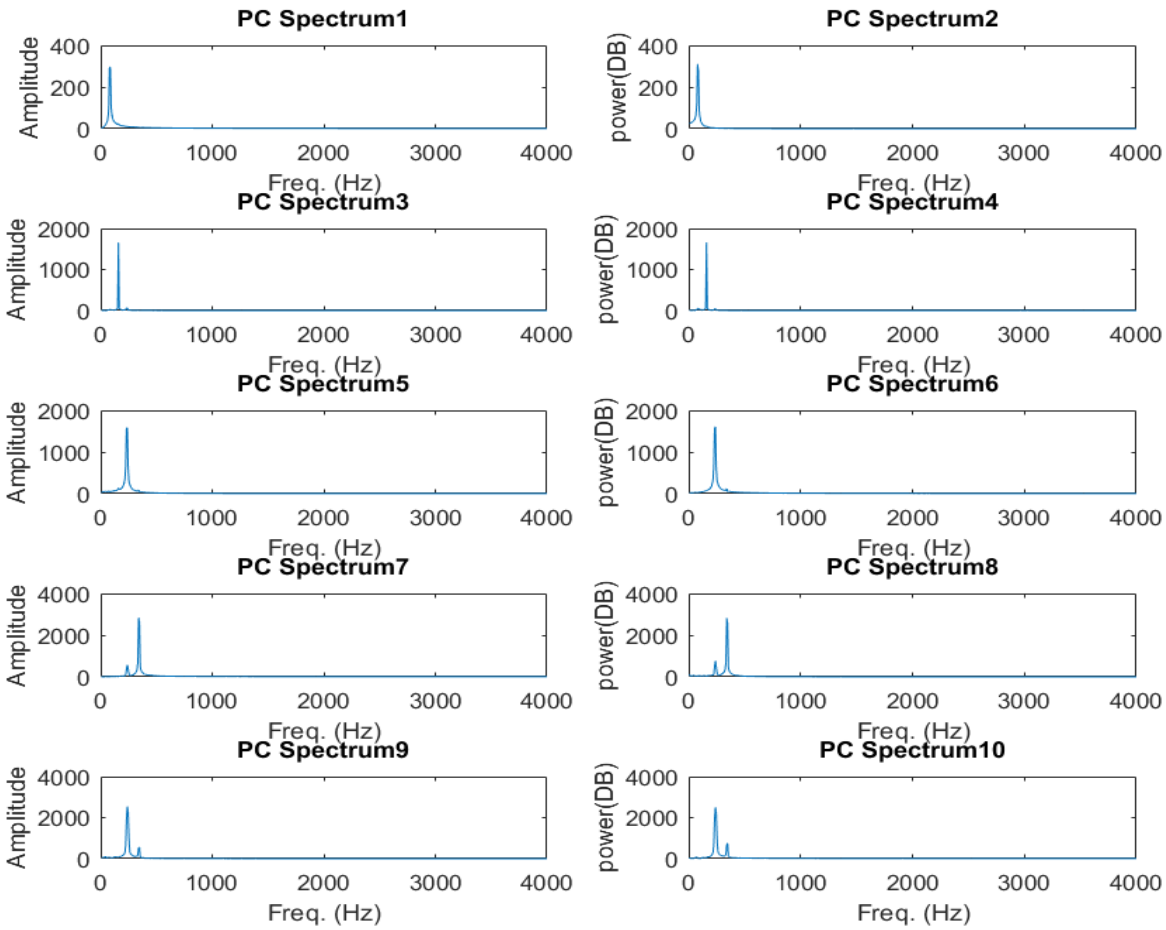


Figure 9-9 Spectrum of the first ten PCs of the given signal illustrated in Figure 8-3; the X-axis indicates frequency and the Y-axis amplitude.

- As mentioned above, each PC can be considered to be a time domain, but on a different coordinate system.

- Mixed sounds (speech and music) are composed of a comb-pattern of harmonics exponentially spaced in frequency. The lowest level of overlapping in the pattern oscillation (frequency) is presented in the PCs and corresponding elementary matrices. Hence, summing a number of elementary matrices together could increase the overlapping level of the contents.

- Furthermore, grouping criterion may lead to error. For instance, grouping PCs corresponding to speech and music with almost equal oscillation together will have an undesirable effect on classification performance.

- The goal is to recognise the contents of soundtracks even where they overlap, with no prior knowledge of the source and minimising the probability of error.
- Drawing on the first two assumptions, and in order to reduce the classification difficulties in the other set of conclusions, this study suggests that each PC of the mixed soundtracks might be classified separately.

9.3.1 The Proposed Classification Method

Drawing on the foregoing findings, it is posited here that the calculated PCs could be used instead of the reconstructed time domain to classify each frame into speech or music based on the extracted features from the corresponding PC vectors.

In an attempt to achieve this proposition, the following steps were carried out:

- Calculation of Principal Components (PCs) of the three classes (speech, music, and mixed) separately, as previously illustrated (see Chapter 8).
- Exclusion of the PCs that correspond to the noise floor to avoid misclassify them (see Section 8-5 in the preceding chapter for more details).
- Centring each of the non-noisy PC vector by subtracting the mean value before the feature extraction step.
- Extraction of the whole set of short-time features mentioned earlier (see Chapter 3) from the calculated PCs.
- Feature scaling using mean normalisation of the features matrix (see Equation 6-8), which is essential. For instance, if the Euclidean distance has increased, this will lead to some of the features being dominant on the distance measure due to their range (Eyben, 2016).
- Training of the Random Forests module on the PCs that are corresponding to the pure classes only (speech and music).

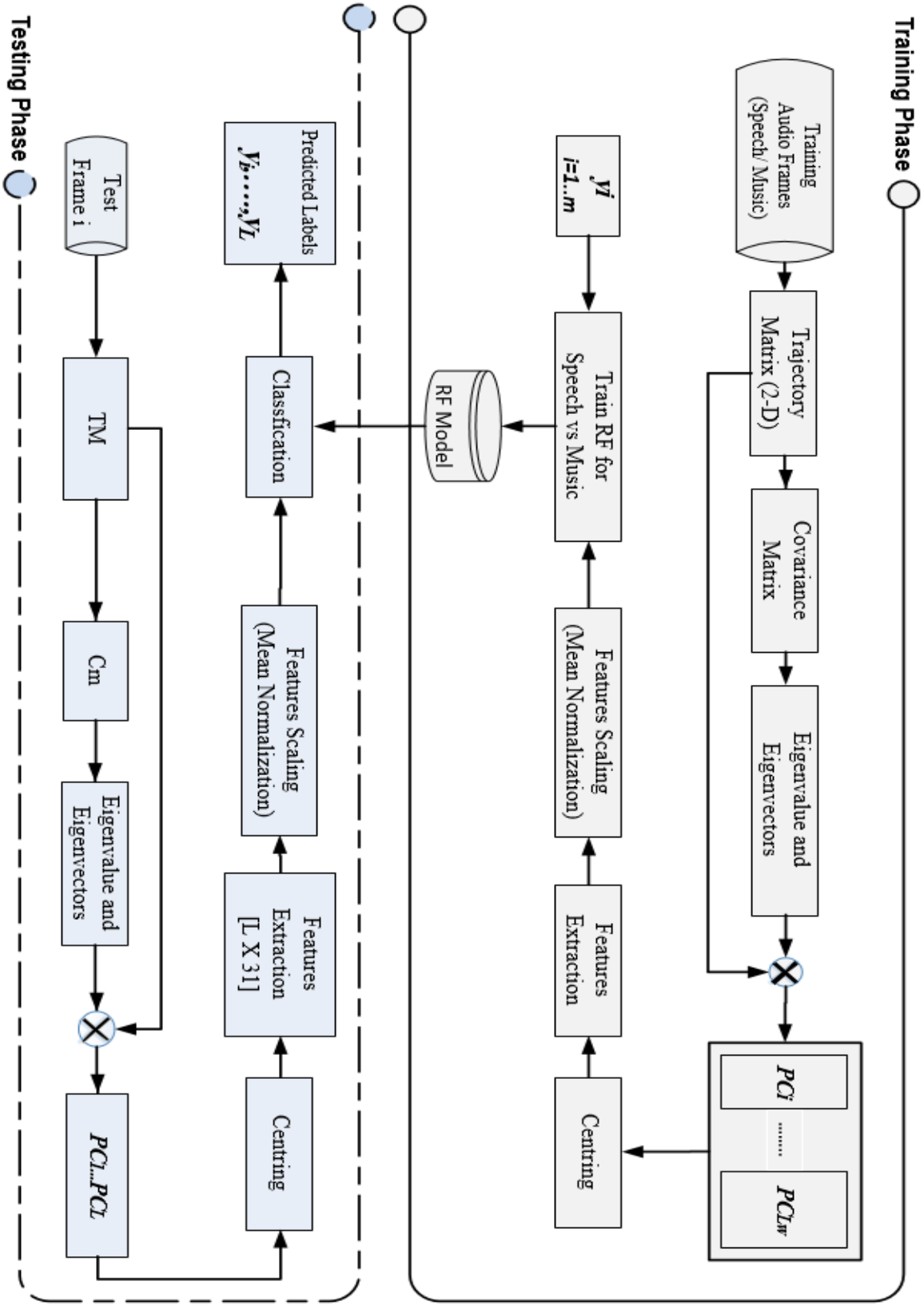


Figure 9-10 General architecture of the RF training and testing phase for PC prediction (Meth.4)

In the testing phase, it is essential to follow the same steps explained above. The flowchart illustrating the procedure for the classification (training and testing phase) of the PCs of mixed soundtracks is shown in Figure 9-10.

9.3.2 Principal Components Calculation

In this method, firstly, the audio samples are processed frame by frame, with a frame size of 100 ms. Next, each frame is mapped into TM with a window length of embedding dimension equal to 30 ms (these length have been optimized, see Section 9.3.7). Then the covariance matrix **Cx** is calculated according to the same method as explained in the preceding chapter, after which the eigenvectors and eigenvalues for the square **Cx** are calculated. The final step in PCs calculation is also the easiest: the desired components are selected with the elimination of the noise floor (see Equation 8-13). To demonstrate the noise removal the largest 25 (lower subspace) eigenvalues (triangular marker) and their corresponding statistical contributions (square marker) are presented in Figure 9-11.

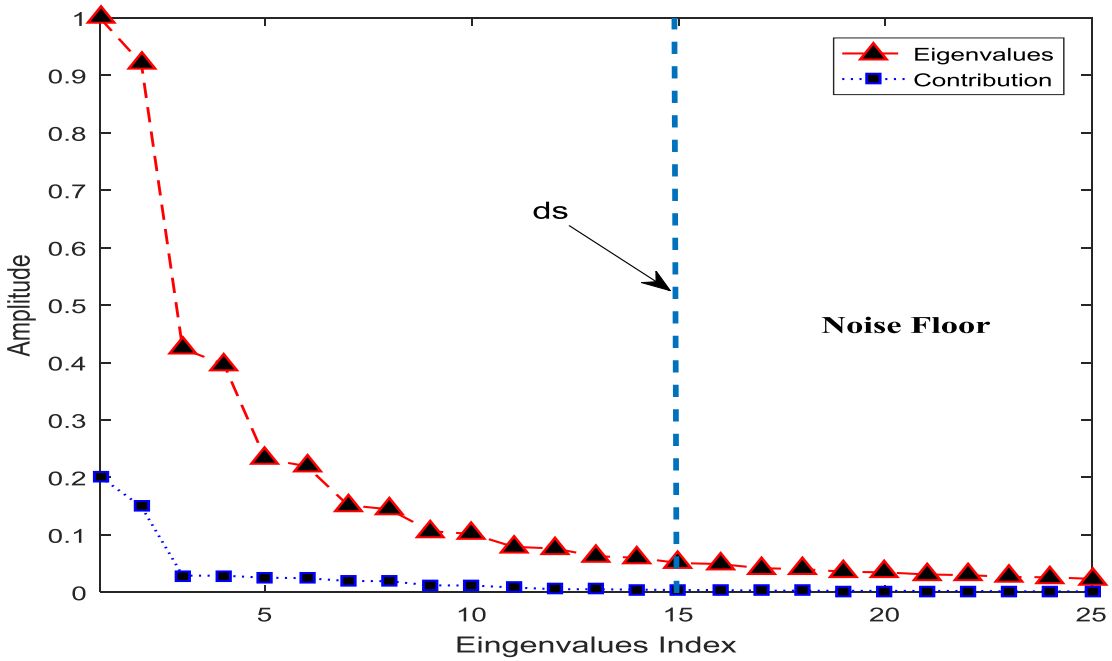


Figure 9-11 Eigenvalues (red) and their contribution (blue) using window length 160

of the audio file presented in Figure 9-1

As demonstrated in Figure 9-11 (contribution line) that PCs after the noise floor have no much information. Consequently, to separate the higher subspace, which contains the noise part of the spectrum, the criterion in Equation 8-13 is applied to omit the noise part with *th* value equal to 0.85 (all PCs corresponding to eigenvalue indexes higher than the ds value are removed). Multiply the transposition of the retained eigenvectors and the eigenvalue itself by the TM, one by one (multiply TM with one eigenvector at a time). (For more information, see Equation 8-10). This will give us the transformed trajectory matrix solely with regard to the selected eigenvectors: these are the PCs.

Figure 9-12 and Figure 9-13 plot the first three PCs in three dimensions for speech and music classes respectively, where the y-axis represents PC2, the x-axis PC1, and the z-axis PC3. As is evident in the graph, the PCs from music have a greater variance than those from speech. It is also possible to clarify datasets with regard to any two or three axes. The representation would be more efficient if the plotted PCs were illustrated in a perpendicular way; this also reflects how important it is to sort the eigenvalues and corresponding eigenvectors in descending order, depending on the variance value from high to low.

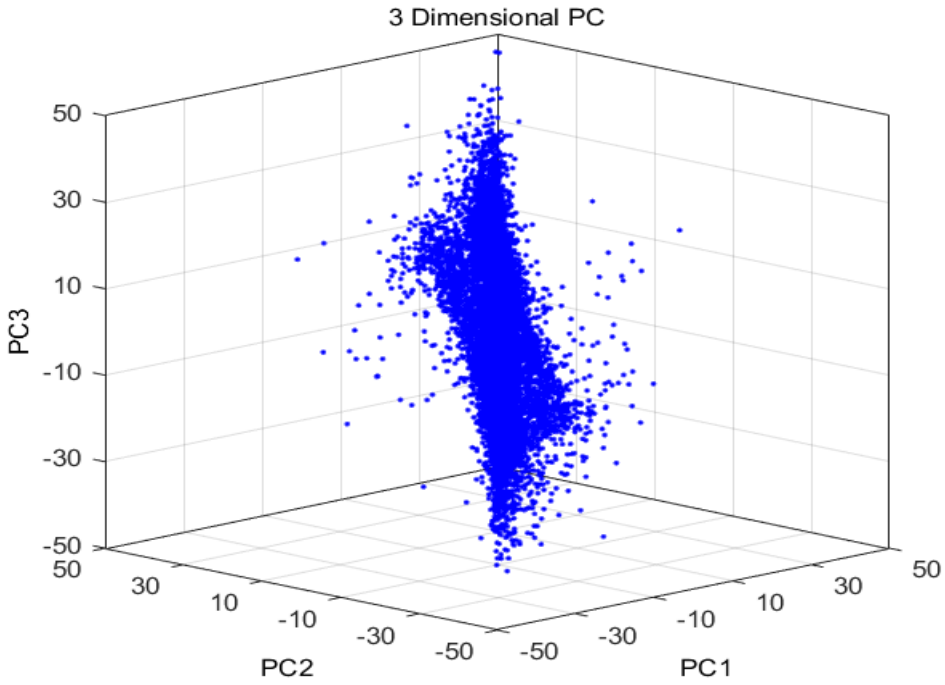


Figure 9-12 the first three Principal Components with the highest variance for speech samples

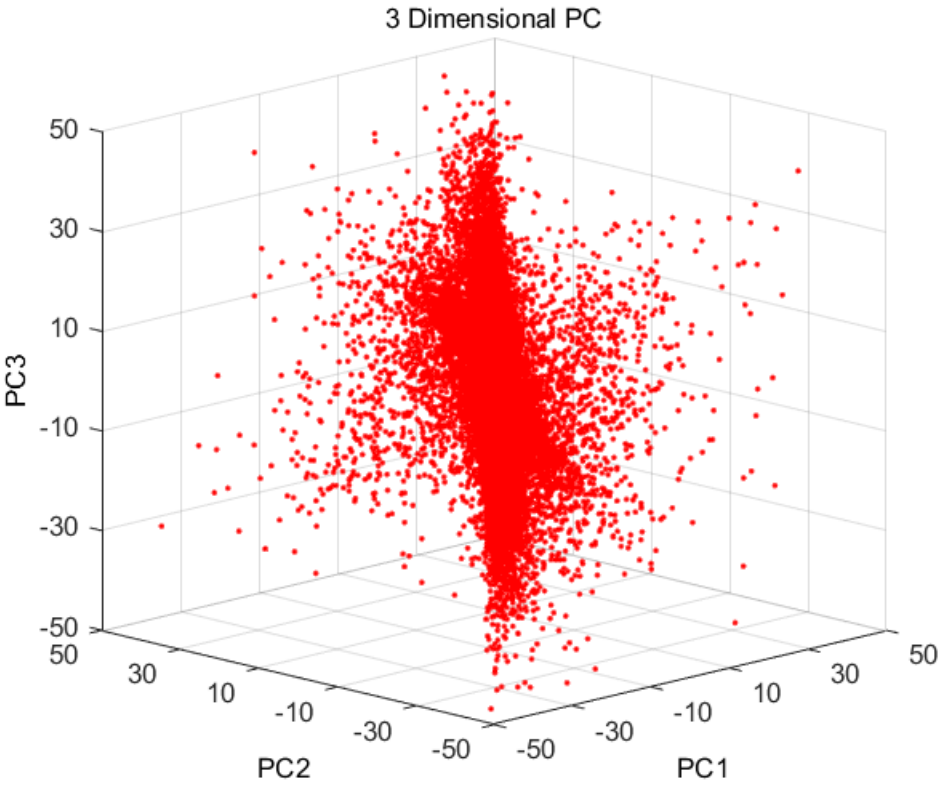


Figure 9-13 the first three Principal Components with the highest variance for music samples

As mentioned previously and demonstrated in the preceding two figures, it is essential to centre each PC vector by subtracting the mean value before the feature extraction step. This is because it has a DC offset (the centre of the PC vector will not be at 0). In other words, the mean of each PC vector must equal zero. This could also be called removing the DC offset, which is considered an undesirable characteristic because it has an effect on the calculation of features including but not limited to ZCR (see Chapter 3). This is performed such that

$$\hat{\mathbf{v}}_j(i) = \mathbf{v}_j(i) - \bar{\mathbf{v}}_j,\ j = 1,...,L_w, i = 1,...,K \qquad 9\text{-}6$$

where $\bar{\mathbf{v}}$ is the statistical average of PC vector, $\hat{\mathbf{v}}_j$ denotes the normalised PC vector, L_w represents the number of principal component (PC) vectors, and K is the length of PC vectors. As depicted in Figure 9-14, the PC centre becomes zero after the DC offset is removed.

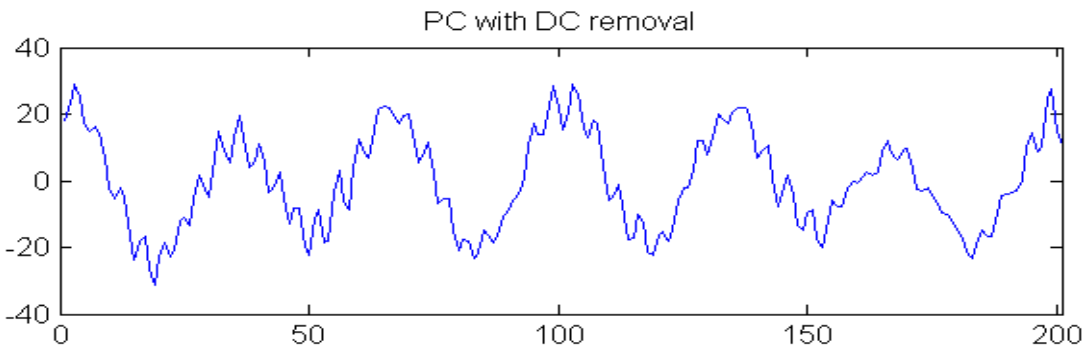


Figure 9-14 PC DC offset removal

9.3.3 Transformed Feature Space

The frames have been transformed into a dataset of PCs expressed as pattern oscillations, which reflect the lines that most closely depict the relationships within the dataset. Consequently, the training dataset of PCs have now been categorised as a grouping or a combination of each particular pattern (lines).

To begin with, number of statistical features are calculated for each PC on a short time-scale, and these features represent a transformed feature space, since they are calculated from a transformed dataset. The feature extraction stage is represented by a two dimension matrix $x = \{m \times Nof\}$, where m denotes the number of the retained PCs and Nof is the number of the calculated PCs. Notably, each frame will be decomposed and then classified based upon the number of the retained PCs. Thus, the feature space **x** is computed for each PC series. Figure 9-15 depicts the transformed ZCR feature, which is extracted from calculated PCs. As demonstrated in the figure, the statistical mean of ZCR (0.2080) for speech is higher than that extracted from music (0.0934), which supports the proposition suggested above.

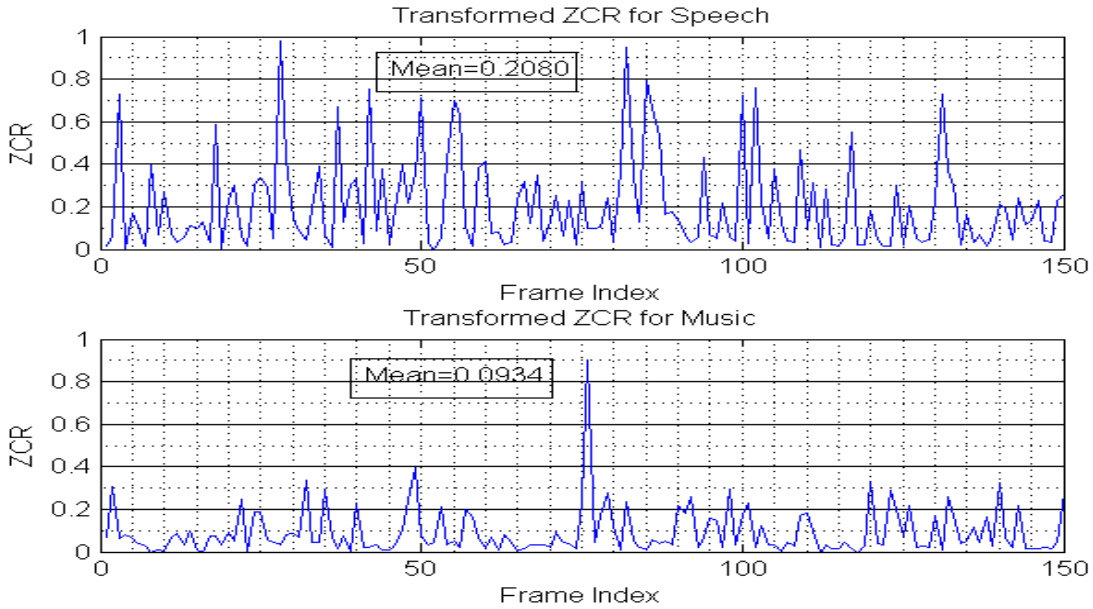


Figure 9-15 Transformed ZCR Feature: ZCR for speech frames (top) and music samples (bottom)

9.3.4 Principal Components Classification

Ultimately, the random forests module was trained as binary classifier for the task of distinguishing speech from music. As described above, during the training phase, the classifier module learns the training set, which is represented by the transformed feature

vectors and corresponding target labels. In this case, the binary classifier was developed with labels 1 for speech and 0 for non-speech (the classifier learns on pure transformed features extracted from the pure classes representing speech and music). Once the RFs module is trained using the training dataset, it can be used to predict the labels of the unseen dataset (unknown data). From the flowchart in Figure 9-16, which illustrates the testing phase, it is apparent that the suggested algorithm adopts voting as its aggregation scheme for the classification of a frame into either speech, music, or a combination of the two.

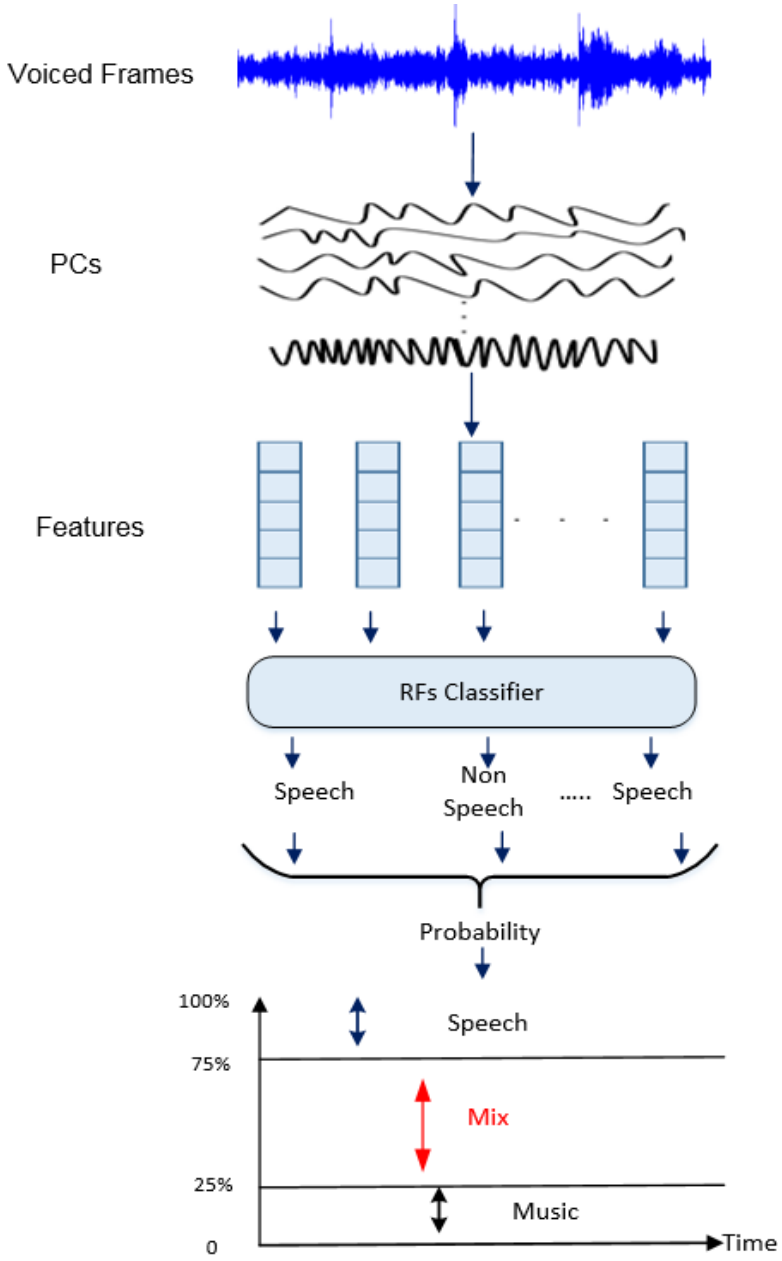


Figure 9-16 general architecture of the mixed soundtrack classification

The probability can be identified with reference to Equation 9-7

$$K=\begin{Bmatrix} Pr(x/c_i)\geq 0.75 & y_i=1\rightarrow(speech) \\ 0.75>Pr\left(x/c_i\right)>0.25 & y_i=0\rightarrow(mix) \\ Pr(x/c_i)\leq 0.25 & y_i=-1\rightarrow(music) \end{Bmatrix} \quad 9\text{-}7$$

K here reflects the aggregation voting for the predicted class (final decision), $Pr(x/c_i)$ is the probability of PCi being classified as speech and can be defined using Equation 9-8:

$$Pr(x / c_i) = \frac{Tp}{\sum Total\_Population} \quad 9\text{-}8$$

where *Tp* represents the true positive (number of classes classified as speech (1)). It is worth noting that mixed soundtracks should be represented by two portions of the set of PCs, one belonging to speech and the other to music.

9.3.5 Performance Measure

In order to evaluate the efficiency of the PC classification in identifying the contents of soundtracks even in the presence of overlapping, RFs classifier for speech, music, or a combination are tested. The sensitivity of a test is identified for each class as the percentage of classified frames allocated to class *c* that indeed belong to class *c*. For instance, the sensitivity of speech is the percentage of frames classified as speech as a fraction of the total number of speech samples in the test dataset. The second class-specific performance measure is precision, or Positive Predictive Value (*PPV*), which denotes the proportion of frames correctly classified as class *c* as a fraction of all the samples classified as class *c*. For example, the *PPV* for the music class represents the number of samples correctly classified as music compared to the total number classified as music. Hence, a better classifier is characterised by a low error rate and higher positive predictive value and sensitivity. Finally, a common performance measure used to make comparisons between more than one classifier, called the F_1 score, is calculated. Table 9-1 presents the confusion matrix for the normalised average of the 10-fold testing. For instance, cell (2, 2) is the percentage of speech frames that were indeed classified as speech following the algorithm described in Figure 9-10, while cells (2,3) and (2,4) show the percentage of speech frames that were classified as music and mix, respectively.

Table 9-1 Normalised Confusion Matrix for the Speech, Music and Mix classification task (k-fold cross validation method, k = 10). (Numbers in %)

Predicted / Actual	Speech	Music	Mix
Speech	95.57%	3.22%	1.21%
Music	3.73%	93.20%	3.07%
Mix	5.40%	5.09%	89.51%
Performance Measurements (% per class)			
Recall	95.57%	93.20%	89.51%
Precision	91.54%	91.81%	90.45%
F_1 Score	93.51%	92.50%	89.98%

As can be seen, there is a significant recognition result. The results also show that features can indeed be extracted from PCs, and reflects a promising improvement in the classification of overlapping classes.

9.3.6 The Optimization of The Frame length and SSA Window length

The relation between frame duration and SSA window length is interesting because the feature space should be extracted from a window that contains sufficient information. Since the frame is mapped into a two-dimensional matrix, and then the last matrix is used to calculate PC vectors prior to computing the feature space for these vectors, both the frame and the SSA length could affect classification accuracy. Consequently, in this study, empirical methods have been used to determine the optimal sizes of these two parameters and via the classification of pure speech vs. pure music datasets. Different frame sizes from 30ms to 100ms have been used, and with each of these frame sizes, different SSA window lengths are deployed in the calculation of PCs. The finding was that the accuracy was increased by increasing the duration of the frame, but was declined by increasing the SSA window longer than the half of the frame length. Using the method of calculating optimal embedding dimension length recommended in Section 9.2.2 and Equation 9-3, the classification accuracy is calculated with different

frame sizes (30 ms-100 ms) in steps of 10 ms, and for each frame size, a different SSA window length is applied. The results, plotted in Figure 9-17, suggest that the size of the frame should be longer than 90 ms, while the corresponding SSA window length should be between $\frac{Frame\_size}{4} \geq L_w \leq \frac{Frame\_size}{2}$ to achieve satisfactory results.

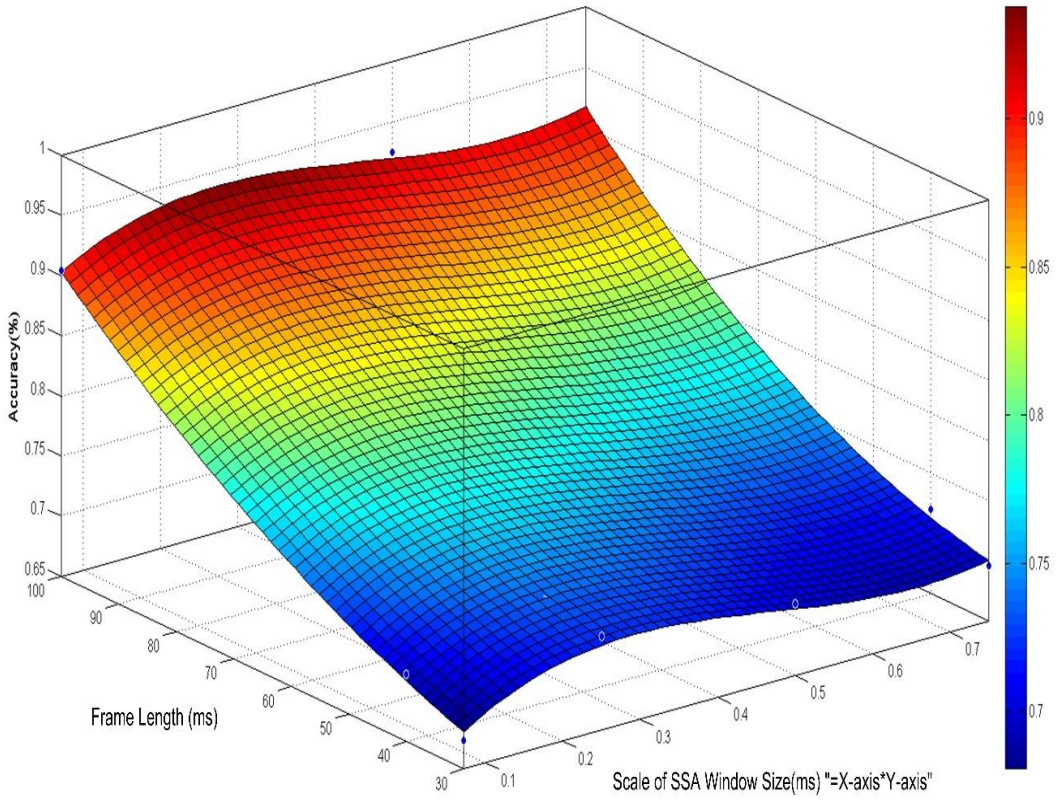


Figure 9-17 Relation between frame size and SSA window length based on the classification accuracy of Meth.4. The y-axis represents frame length, the x-axis SSA window length, and the z-axis accuracy.

9.4 Summary

Two different methods were conducted to investigate the ability of SSA to decompose mixed soundtracks into different components with mostly non-overlapping content. In the first method, a number of audio features were extracted. To this end, random forests decision trees were employed to classify the enhanced output signals from SSA into silence, speech, music, or a combination thereof. An alternative method was also sug-

gested and applied. Instead of reconstructing the filtered components into the time domain and then classifying them, the features that extracted from PCs were fed directly into RFs to classify them as speech or non-speech. Finally, the probability of PCs being classified as speech was used as a voting criterion to classify the corresponding frame into speech, music or a mix.

The proper grouping criterion, SSA window length and frame size for both of the developed methods is optimized and validated as explained before in this Chapter. The next Chapter will demonstrate the results of the suggested methods in this study, comparison between these sets of results and results discussions.

10 RESULTS AND COMPARISONS

10.1 Overall System Testing

A number of experiments were conducted on the proposed benchmark database with 11 levels of different mixing ratios in a methodical way as shown in Figure 6-2. All pure and mixed samples are manually labelled after the framing stage into speech, music or mix. To evaluate the proposed methods the 10-fold validation method has been applied. Hence, each mixing group was separated in a random way into 10-subsets and then the suggested experiments were conducted for 10 iterations. At each iteration, nine subsets were used for training and the remaining subset for testing.

With each of the first two suggested methods (Meth.1, Meth.2, see Chapter 4 and Chapter 7), two modules were trained, one for speech and the other for music. Each of those two modules trained on detection of its respective class through including all samples with the presence of that class against all others. For instance, in the case of the speech classifier the pure speech and mixed samples are trained against pure music. While the music classifier is trained on music and mixed samples against pure speech. As mentioned earlier, each method has trained with a different set of feature space. These two sets of feature space were illustrated in Chapter 3 and Chapter 7, which are a raw short-term feature space and the augmented feature space from mid-time (almost one second). Consequently, four groups of results will be presented to evaluate the implemented methods (Meth.1, Meth.2, Meth.3 and Meth.4) besides the results in Chapter 6, which represent the evaluation of MARSYAS and a pilot case study.

In contrast, in the last method (Meth.4) only one classifier has been trained for detection of pure speech, music, and mix due to decomposing each frame into a number of PCs as explained before. Hence, each PC will be classified as either speech or music; thereby

each PC will represent an individual decision. Subsequently, the nature of the frame contents can be determined by voting for the class with the highest probability; otherwise if the probability is between 25% and 75%, then the frame will be classified as a mix of speech and music, see Equation 9-7 and Figure 9-16 (for more details).

To inspect the efficiency of SSA in the improvement of the classification intelligence of mixed soundtracks, RFs classifiers for speech, music, or a combination of thereof are tested. Random Forest classifier (Section 4.6.2) is the choice for all conducted experiments with 1000 trees because it provides both adequate results (from the literature study) and the ability to manage a big dataset. As mentioned, the order of samples in the training subset (not in the test partition) was randomised using the shuffling method. Moreover, all extracted feature vectors were normalised to the mean.

It is important to identify the measurement of the results that applied in this research, as follows:

Tp: True positive, refers to the number of the correctly classified frames. Hence, the *Tpr* of a test is identified per-class as the percentage of frames classified to class *c* and that correctly had the class label *c*. For instance, the *Tpr* of speech is the percentage of frames that were classified as speech as a fraction of the total number of speech samples in the test dataset. It is also known as a sensitivity or recall and is as given in Equation 10-1

$$Tpr_i, Re_i = \frac{Correctly\_Classified\_as\_Positive}{Total\_Postive\_Samples} = \frac{Tp_i}{Tp_i + Fn_i} \qquad 10\text{-}1$$

Tn: denotes the correctly rejected frames.

Fp: the incorrectly accepted frames. For instance, music frames which areincorrectly accepted as speech frames in the case of music classifier.

Fn: False negative, incorrectly rejected frames.

PPV: Positive Predictive Value denotes the proportion of frames that are correctly classified as class *c* out of all frames classified as class *c*. It is also known as the Precision (*Pr*). Equation 10-2 gives it as

$$Pr_i, PPV_i = \frac{Tp_i}{Tp_i + Fp_i} \tag{10-2}$$

The accuracy (overall, how often is the classifier correct) can be defined as given in Equation 10-3.

$$Accuracy = \frac{Tp + Tn}{\sum \text{Total population}} \tag{10-3}$$

The performance estimation is Unweighted Average Recall (*UAR*) (Schuller, 2013) which could be considered as a preferable indicator of the accuracy over all classes (true performance) of the baseline and proposed modules for an appraisal where the appearance of class labels is extremely unbalanced across the different classes. Standard accuracy also refers to the UAR indicator, and it is calculated as illustrated in Equation 10-4 (Schuller, 2013):

$$UAR = \frac{1}{c}\sum_{i}^{c} Re_i \tag{10-4}$$

where *c* represents the total number of classes (in this study $c = 2$, speech and music) and Re_i denotes the recall portion for the i^{th} class. In the same way, Unweighted Average Positive predictive (*UAP*) value has been calculated as given:

$$UAP = \frac{1}{c}\sum_{i}^{c} Pr_i \tag{10-5}$$

where Pr_i reflects the precision for the i^{th} class, therefore the classifier performance is

given by F_1 score, which can be calculated as given in Equation 10-6

$$F_{1,i} = 2\frac{Re_i Pr_i}{Re_i + Pr_i} \quad \text{10-6}$$

F_1 score is a broadly known performance measure for comparison between more than one classifier which is called the F_1 score is used and calculated.

Finally, Unweighted average F_1 (UF_1) can be defined as given in Equation 10-7

$$UF_1 = \frac{1}{c}\sum_{i=1}^{c} F_{1,i} \quad \text{10-7}$$

UF_1 Score is calculated for the baseline classifier Meth.1 explained in Chapter 6 and for the other suggested methods (Meth.2, Meth.3, and Meth.4) as a comparison measurement. Moreover, speech mixed with music in multiple speech-to-music ratios ranging from -20 dB to 20 dB in steps of five are used to validate the proposed method. For comparison between the classifiers of the suggested methods, UF_1 score as given in Equation 10-7 is used and calculated.

10.2 Results Comparisons

The measurements depicted in Table 10-1 below demonstrate the normalised average of the 10-fold cross validation technique, which is applied for training and testing of the modules. The overall performance of the classifier is evaluated by taking the average of the ten folds. As can be seen in Table 10-1, two classifiers have been developed with Meth.1 - one classifier for recognising speech against all and the second classifier for music against all. Additionally, the performance average of both classifiers is offered in the 'Performance Average' row. This performance is the UF_1 of both classifiers. As can be seen, the baseline system Meth.1 works well with pure classes (non-overlapping

condition between speech and music), but the results were not adequate where the overlapping increased between speech and music. In most mixing ratios, greater than 25% of frames were misclassified, either *Fp* or as *Fn*. This observation has a physical meaning since the overlapping contaminate the characteristics and features and causing poor classification performance. Another reason is that existing features for classification are predominantly established on artificially tailored, non-overlapping audio clips, which make these features, which extracted from short period, insufficient to determine the overlapped contents.

Table 10-1 Classification Recall and PPV and Unbalanced F1 Score of Meth.1, M denotes pure music and Sp refers to pure speech, and the other values denote the mixing ratio in dB

		Actual/ Mixing Ratio										
Exp.1	Measure	M	-20	-15	-10	-5	0	5	10	15	20	Sp
Speech Classifier	Recall %		75.82	83.64	80.23	83.28	77.07	78.62	78.33	91.66	94.75	94.97
	PPV %		69.18	63.05	61.67	67.78	67.34	65.81	76.83	79.00	79.53	87.32
	F1 %		71.88	72.06	69.62	74.63	71.84	72.04	77.57	84.86	86.48	90.98
Music Classifier	Recall %	89.96	84.21	87.77	88.22	86.15	74.33	74.42	74.27	75.48	75.61	
	PPV %	86.39	80.13	84.49	79.79	75.78	71.94	71.11	73.77	70.14	74.68	
	F1 %	88.16	82.12	86.11	83.61	80.68	73.12	72.22	74.02	72.71	75.14	
Performance Average	UF1 %	88.16	77.00	79.08	76.61	77.65	72.48	72.13	75.80	78.79	80.81	90.98

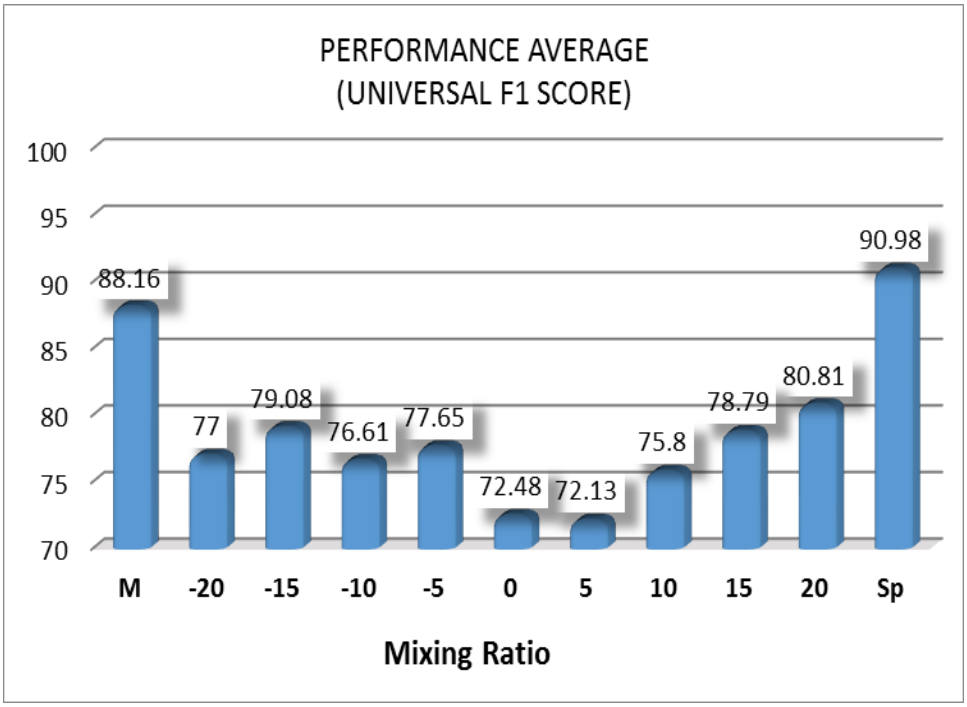


Figure 10-1 Performance Average (F1 Score %) for Meth. 1

The same measurements were used with the other Meth. 2 shows the UF_1 score results of using mid-time statistics feature space (1-second) and their measurements relative to the speech and music classifier. The row 'Performance Average' contains the results of the normalised average F_1 score of both classifiers calculated using Equation 10-7. In particular, using mid-time feature space, the harmonics of the music frames behave to determine the occurrences of the music within both classifiers slightly more accurately than the results that can be achieved using the raw feature space that extracted from 50 ms frames and demonstrated in Table 10-1. This seems to suggest that giving more information through applied the suggested augmented features with longer window over time period can slightly improve the classification performance, which is not surprising.

Table 10-2 Classification Recall and PPV and Unbalanced F1 Score of Meth.2, M denotes pure music and Speech refers to pure speech, and the other values denote the mixing ratio in dB

		Actual/ Mixing Ratio										
Exp.2	Measure	M	-20	-15	-10	-5	0	5	10	15	20	Sp
Speech Classifier	Recall %		77.31	71.33	71.36	66.30	85.66	88.45	84.59	89.00	84.38	89.82
	PPV %		61.97	69.83	63.35	56.00	62.62	75.08	78.77	64.00	87.23	85.49
	F1 %		69.56	69.37	66.82	60.72	73.17	81.63	81.06	75.76	85.78	87.60
Music Classifier	Recall %	93.43	90.50	87.60	87.12	90.10	86.70	84.57	89.49	84.63	88.89	
	PPV %	91.70	89.60	89.13	86.78	89.23	83.97	85.81	75.70	75.95	71.84	
	F1 %	92.56	90.05	88.36	86.95	89.66	85.31	85.19	82.48	79.89	80.34	
Performance Average	UF1 %	92.56	79.80	78.86	76.89	75.19	79.24	83.41	81.77	77.83	83.06	87.60

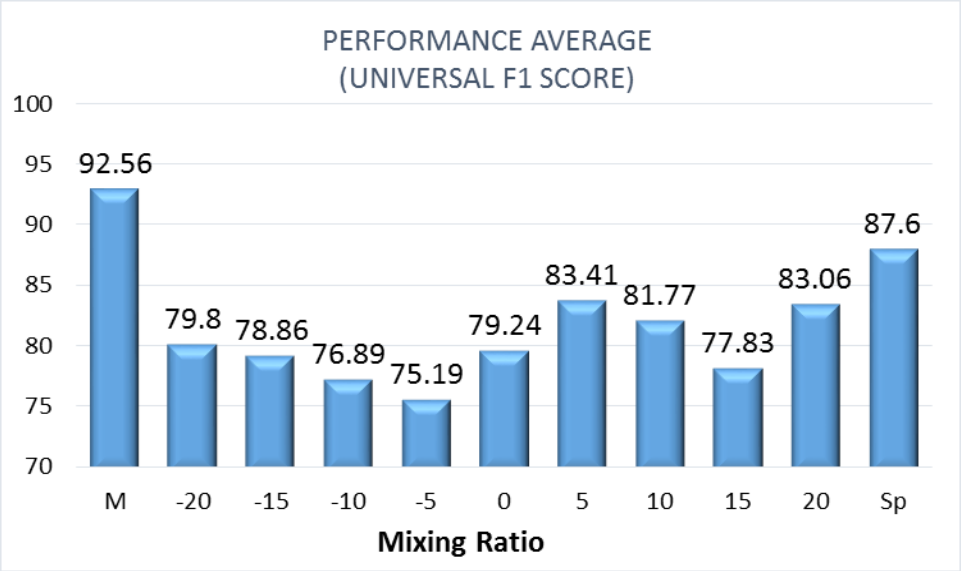


Figure 10-2 Performance Average (F_1 Score %) for Meth.2

Data from the foregoing two tables (Table 10-1 and 10-2) can be compared with the

data in Table 10-3, which shows significant improvement in the sensitivity, precision, and thereby the accuracy of the classification, for all mixing levels. In fact, the improvement was up to 20% of the value of the F_1 score, as shown in the two tables below (Table 10-3 and 10-4), which correspond to the SSA technique (Meth.3 and Meth.4), explained in Chapter 9. This significant improvement using Meth.3 can be conceptualised as the SSA enhanced the classification performance through mitigating the overlapping between the components by separating them into a two time series with a lower ratio of overlapping and then classifying them separately.

Table 10-3 Classification Recall and PPV and Unbalanced F1 Score of Meth.3, M denotes the pure music and Sp refers to pure speech, and the other values denote the mixing ratio in dB

		Actual/ Mixing Ratio										
Exp.3	Measure	M	-20	-15	-10	-5	0	5	10	15	20	Sp
Speech Classifier	Recall		90.13	89.83	93.43	93.20	92.72	92.31	94.76	96.89	91.71	94.30
	PPV		88.93	90.74	91.27	94.06	95.12	95.47	89.77	84.16	94.55	91.03
	F1		89.53	90.28	92.34	93.63	93.90	93.87	92.20	90.08	93.11	92.64
Music Classifier	Recall	94.73	94.77	86.90	94.77	92.81	91.07	98.57	87.25	91.85	88.88	
	PPV	91.35	91.89	90.31	92.56	89.96	93.80	85.18	93.22	94.62	86.64	
	F1	93.01	93.31	88.57	93.65	91.36	92.41	91.39	90.14	93.21	87.74	
Performance Average	UF1	93.01	91.42	89.43	92.99	92.50	93.16	92.63	91.17	91.65	90.43	92.64

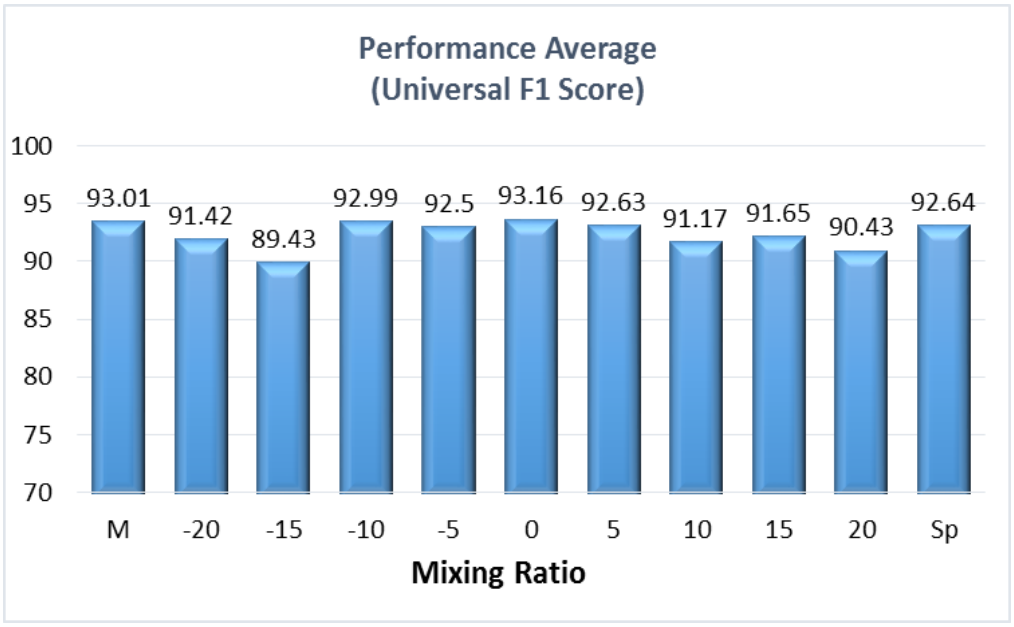


Figure 10-3 Performance Average (F1 Score %) for Meth.3 (different mixing ratios)

Finally, Table 10-4 shows the results of the proposed method 4 (Meth.4) that used classification based on the transformed features extracted directly from the PCs. As demonstrated, the proposed method provided a power classification performance and outperformed of both the Meth.1 and Meth.2, since each PC corresponding to particular oscillation, which is represented a lower level of overlapping ratio between audio sources, and each oscillation represented by particular PC is classified separately. However, the results were somewhat lower than the results of the previous method (Meth.3) but with much lower performance time. The result is in the lines of earlier discussion (Section 9.3) that conclude that each PC of the mixed soundtracks can be classified separately in order to reduce the classification difficulties.

Table 10-4 classification Recall and PPV and Unbalanced F1 Score of Meth.4.

		Actual/ Mixing Ratio										
Exp.4	Measure	M	-20	-15	-10	-5	0	5	10	15	20	Sp
Classifier Performance Measurements (%)	Recall	90.65	92.27	93.6	91.63	89.83	91.32	92.86	93.14	89.02	94.75	92.92
	PPV	94.1	89.87	91.48	86.51	87.97	90.34	87.26	88.57	87.65	86.41	94.77
	UF1	92.34	91.05	92.53	89.00	88.89	90.83	89.97	90.80	88.33	90.39	93.84

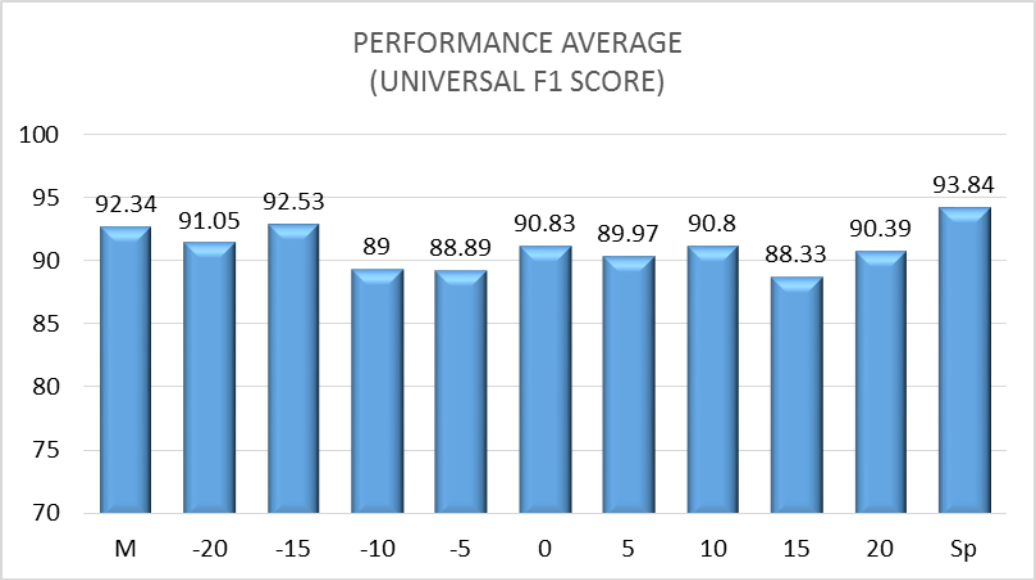


Figure 10-4 Performance Average % of Exp4 for different mixing ratio

The comparisons between all the experiments conducted (Meth.1, Meth.2, Meth.3 and Meth.4) are highlighted in Figure 10-5. The most significant improvement were with Meth.3 and then Meth.4, see Table 10-3. However, the performance time of Meth.3 (reconstruct to the time domain and then perform the classification process) was much longer than the processing time of Meth.4 (classify the frames based on the corresponding PCs). Notably, the latter can determine the content of the audio frames through developing only one classifier. Additionally, the newly proposed and developed method

(Meth.4) can detect pure speech, pure music, and mix with a much lower complexity level.

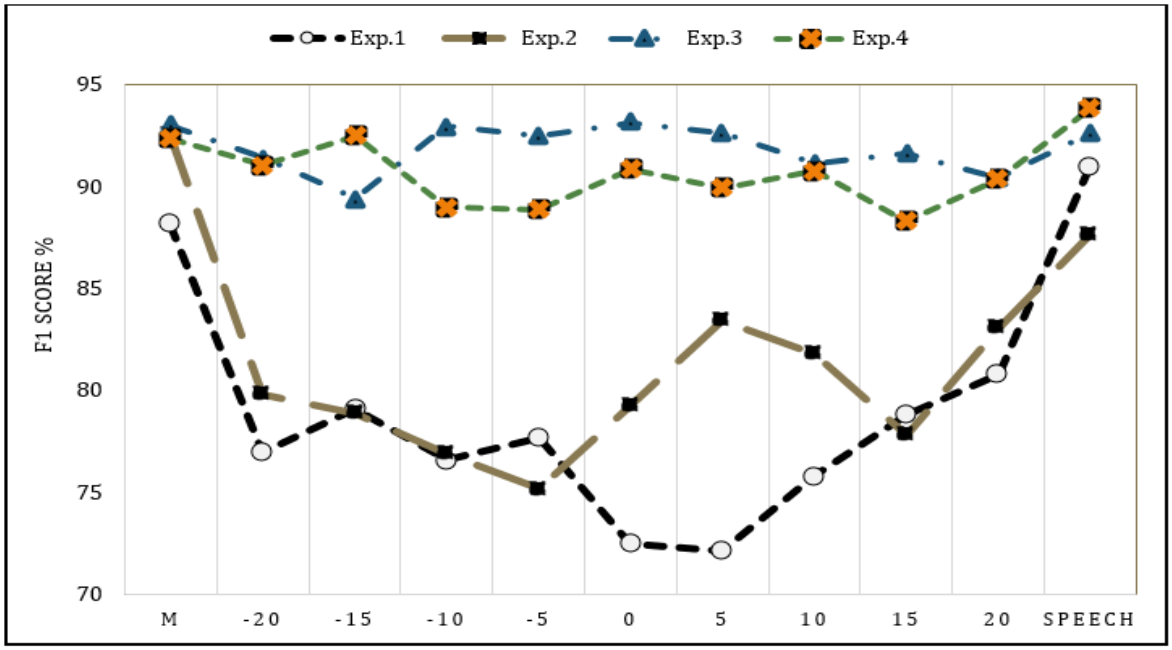


Figure 10-5 Comparison of baseline classification method with all suggested and applied methods. Y-axis is the F1 score percentage, and X-axis depicts the speech-to-music mixing ratio in dBs

In Figure 10-5 there is a clear trend of decreasing the value of the F_1 score for both Meth.1 and Meth.2 when the mixing ratio lies in the mid-range (-15 to 15). By contrast, considerable improvement in the classifier performance is noticed in the recognising of mixed soundtracks using the suggested SSA decomposition technique (Meth.3 or Meth.4). The justification of the results of each experiment is highlighted as discussed above. It is also perceived that mixing ratio has significant effects on the classical classification that presented in (Meth.1) consequent to the undesirable effect on the semantics of the features.

Furthermore, the standard error for each method has been estimated using the following steps:

- Calculate the statistical mean value of UF_1 for all mixing groups in "average performance" row according to the Equation 10-8

$$UF1' = \frac{\sum UF1}{n} \tag{10-8}$$

where (n=11) denotes the number of the mixing data groups.

- Estimate the standard deviation of UF_1, which can be calculated according to Equation 10-9

$$S_{F1} = \sqrt{\frac{\sum \left(UF_1 - UF_1'\right)^2}{n-1}} \tag{10-9}$$

- Estimate the standard error of UF_1 as given by Equation 10-10

$$ESE\left(UF1'\right) = \frac{S_{F1}}{\sqrt{n}} \tag{10-10}$$

- Finally, determine the error boundaries by subtracting the standard error from the mean and recording that number, then adding the standard error to the mean and recording that number. These two values represent the distance from 1 standard error below the mean to 1 standard error above the mean, as shown in Figure 10-6

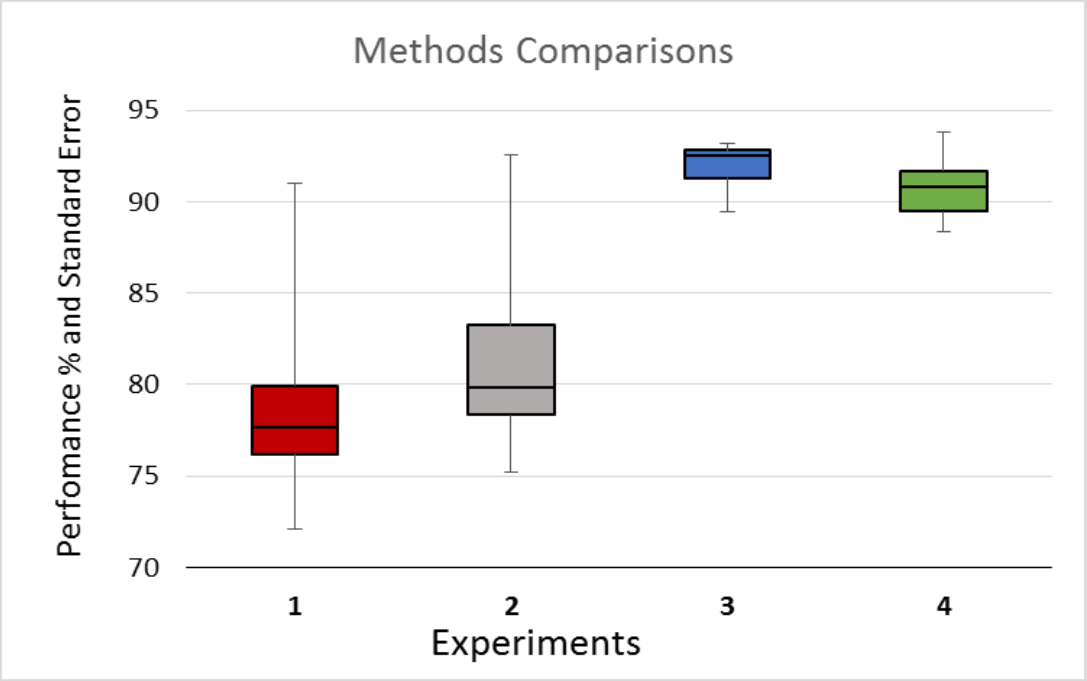


Figure 10-6 Suggested method comparisons using normalised standard error for UF1 score

The central bold line for each experiment is the medium of the distribution over all mixing groups: it is apparent that method Meth. 3 is represented by the highest medium and Meth.1 has a lower medium. The smallest standard error values are represented by Meth.3 and Meth.4 respectively. The whiskers of each experiment refer to the minimum and maximum observations of the statistically averaged value of UF_1 score for each proposed method. It is apparent that SSA methods outperformed the other two methods through providing smaller estimated standard errors and higher UF_1 scores as shown in Figure 10-6.

Finally, Table 10-5 depicts the accuracy in each of the 10 folds of the RFs module with the suggested method in Meth.4. It can be seen from the Table that the random forests technique is very stable and that promising accuracy results are achieved.

Table 10-5 Classification results of Meth.4 accuracy (%) in Ten folds for all mixing ratios, where M denotes pure music and Sp refers to pure speech, and other values denote the mixing ratio in dB The abbreviation Ite. refers to iteration fold.

Ite. / Actual	Ite. 1	Ite. 2	Ite. 3	Ite. 4	Ite. 5	Ite. 6	Ite. 7	Ite. 8	Ite. 9	Ite. 10
M	93.22	91.54	92.75	92.08	87.25	92.31	95.1	92.74	89.22	91.85
-20	89.23	94.51	92.55	86.47	94.51	85	92.55	90.39	94.51	87.92
-15	89.73	91.79	90.78	89.85	90.64	89.63	86.74	89.96	91.58	92.51
0	86.67	88.85	88.63	92.55	86.92	92.55	90.59	89.23	92	90.59
-5	87.47	89.62	94.71	94.9	93.46	87.31	93.46	87.06	87.45	89.41
0	92.75	92.75	86.86	89.62	94.71	85.88	82.35	93.46	86.75	91.71
5	85.38	92.55	93.46	90.93	94.94	91.54	88.24	86.71	91.54	87.86
10	90.98	87.69	88.63	87.69	89.23	92.55	86.86	92.75	88.82	91.54
15	92.38	90.9	91.71	85.38	90.78	86.24	90.38	88.82	90.78	87.38
20	89.41	93.33	88.18	92.31	93.33	90.77	85.49	87.25	88.46	89.02
Sp	90.59	92.55	94.9	93.08	94.62	94.62	90.77	91.6	92.75	90.77

10.3 Evaluation of Usability with Real Worked Samples and Discussion

10.3.1 Evaluation OF Usability with Real Worked Samples

For the validation of method Meth.4 with worked samples, ten samples from each class of the GTZAN dataset (speech and music) with length of 30 seconds are mixed together by one of the audio production masters students at Salford University. The students have gained practical, theoretical, and creative experience in sound engineering, music production and audio technology. Consequently, they have aquired the skills needed to create and deliver professional audio. The mixed samples therefore have similar characteristics to those of real world audio samples with regards to their speech and music content. More details about the audio production lab can be found on the web site (Salford, 2016) and in Appendix B. To listen to the produced audio files, browse the attached CD with Appendix B.

Furthermore, each audio file has framing into (599) frames with length of 100 ms. Then, each frame is decomposed into a number of PCs equal to 560. The pre-training modules on the pure classes were used to classify the mixed soundtracks illustrated in the preceding paragraph. The statistically averaged values of K-voting over all frames are

presented in

Table 10-6.

Table 10-6 Accuracy of mixed soundtrack classification

File ID	Average of 10-folds
File 01	83.17%
File 02	84.62%
File 03	82.57%
File 04	84.70%
File 05	85.37%
File 06	80.82%
File 07	85.64%
File 08	86.82%
File 09	84.50%
File 10	83.94%

Figure 11-1 shows the standard devaision boxplot of the calculated average of accuracy over all mixed soundtracks. The line inside the boxplot refers to the medium value over all files, while the whiskers refer to the minimum and maximum observations of the statistically averaged value of accuracy. The boxplot indicates that the accuracy distribution is almost normal. (This data is positively skewed; that is, there are more observations towards the higher end of the boxplot).

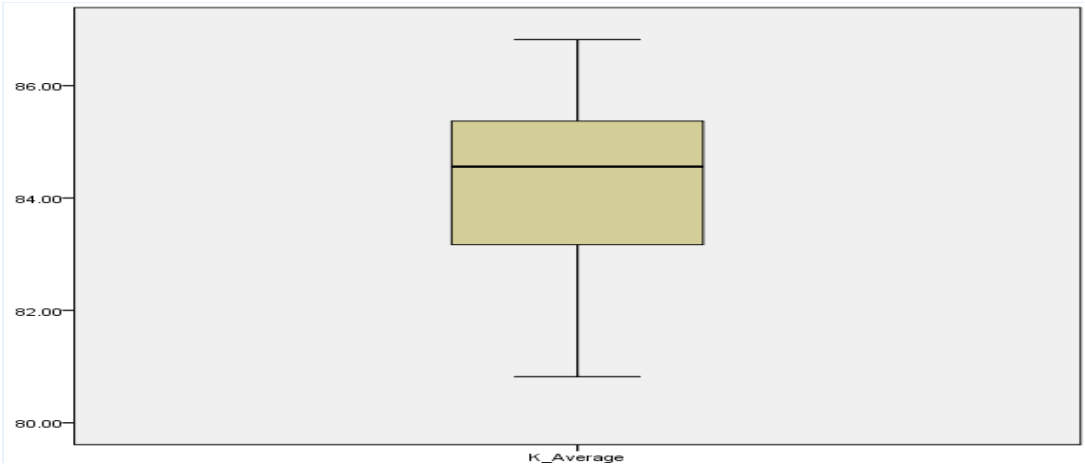


Figure 10-7 Standard Deviation of 10-average value over all tested mixed files

When the trained system is exposed to a different dataset produced professionally in production studio, as shown in Appendix B, it shows a good performance even though small differences are documented because the professional production sound involves a different compression and equalization level.

10.3.2 Discussion

An extensive approach is required to provide an efficacious classification of mixed soundtracks. Initial results in these areas have been reported, and critical challenges remain. In this research, a novel method to identify overlapping sources in mixed soundtracks employing the SSA technique is proposed and conducted. With a view to assessing the suggested method, speech and music were collected from the GTZAN database and mixed in different speech-music ratios. Hence, the suggested method is compared with conventional classification methods using a state of the art machine learning technique.

The choice of the GTZAN dataset for this work was as a result of its use in many publications and PhD projects mentioned in the literature for speech and music genre categorization. Furthermore, it supports the generalisation condition, e.g. it provides speech samples in different languages and for both genders and for most music genres.

The classification of mixed soundtracks combined with SSA requires criterion of decomposition of the mixed sources into different independent eigenelements of the input leads, which is proposed in this research. As explained, two different methods for enhancing the classification results of mixed soundtracks applying the SSA technique have been suggested and studied. Theoretically, the last method (Meth.4) should offer the most improvements in soundtrack classification since each PC corresponding to particular oscillation, which is represented a lower level of overlapping ratio between audio

sources, is classified separately. Therefore, more investigation is required to evaluate the proposed algorithm and to indicate the most appropriate feature space. Also, the RFs can be learned on the PCs directly without the need to calculate features for them. This suggestion is supportive by Figure 9-12 and Figure 9-13, which illustrate that each of speech and music has a particular pattern.

Results demonstrated in the preceding chapter are obtained by processing the audio samples in frames of 50 ms, 100 ms, and 1 second. In this study, the researcher notes along with others that a 50 ms frame length could be a reasonable size for feature space extraction, but with the suggested method (Meth.4), the reasonable size would be longer than 90 ms due to decomposing the frame into smaller windows (PCs) and then extracting feature space from the decomposing PCs.

The performance of Meth.3 (mitigating the overlapping between the mixed sources through SSA algorithm processing and then reconstructing the grouped oscillations into the time domain and extracting the feature space from the enhanced pair of outputs and then classifying them) was slightly better than Meth.4 (which extracted features from PCs and then classified them directly without reconstructing the entirety back to the time domain). Nevertheless, that did encounter a few obstructions, including excessively long processing time, increased storage requirements (each frame symbolised by two outputs), and this all leads to greater computational load than previously. In conclusion, this study indicates that the SSA techniques could be applied for improvements in the classification of mixed soundtracks. The study provides comparisons indicating that the suggested algorithms outperformed the conventional classification methods in the presence of overlapping between speech and music. For the PCs classification method, the suggested method provides significant performance with simple and low computational load.

11 CONCLUSION AND FUTURE WORK

11.1 Conclusion

The research has been focused on mitigating the classification difficulties associated with overlapping between speech and music classes and improving the classification of the mixed soundtrack. For evaluation purpose, a simulated dataset generated by the mixing of pure speech samples with pure music samples related to different levels of the speech-to-music ratio is established and examined.

The study started from a pilot investigation into the modification of a well-developed system namely MARSYAS with the aim to improve its capability of handling overlapped content. This has shown to improve classification accuracy in the case of overlapped audio with speech and music from virtual nil to over 60% for speech detection and over 76% for music. However, for typical audio classification without overlap, a performance circa 90% accuracy is generally achievable; therefore, higher accuracy in the overlapped cases is sought after. The study then proceeded to the employment of two binary classifiers simultaneously using the random forests technique, one for speech and the other for music. The target of each classifier is to determine the occurrences of the corresponding class. The results of this investigation show that the accuracy is over 72% when the overlapping takes place between the audio content represented by all overlapped mixing levels, as shown in Table 10-1, while, the classification accuracy represented by the F1 score measurement was over 90% for both speech and music without overlapping. Moreover, analysis of the computed features, consistent with other studies highlighted in the literature review chapter, shows the following: 1) A long window could maximise the extracted amount of information from the analytical

frame. 2) The target pattern can be detected by the behaviour corresponding to the number of analytical frames. The research then moved forward to investigate the use of augmented and medium term features with random forests to tackle the overlapping problem. From the results in Figure 10-5, it is apparent that there is slight improvement in the detection of overlapped contents to over 75% using the augmented features. This seems that the technical challenge in handling overlapped audio classes remains unsolved. Finally, for mitigating the classification difficulties of this problem, two new methods using SSA are developed, one for eliminating overlapping in mixed soundtracks and the second for PC classification by applying transformed features calculated with SSA.

The main two parameters of SSA method, the window length of embedding dimension and the grouping criterion, were examined in this study and the optimal dimension and the proper idea are suggested and then validated. Accuracy and classifier performance of both overlapped source detection after processing with SSA is over 89 %, even for signals with 0 dB (equal level of overlapping between speech and music). High classification accuracy and positive predictive values of decomposed signals are documented. Consequently, over 17% improvement in the sensitivity of decomposed signals for the speech-to-music ratio of 0 dB has been achieved. It is worth noting here that the computational load of the proposed method is increased due to the extra mathematical complexity of the SSA algorithm and the deployment of two classifiers as a part of the proposed non-exclusive classification method. This extra computational load is circumvented in the subsequently proposed method.

Since TM component of the singular value decomposition is totally identified by the respective PCs, then the classifier could be used the PCs for speech/music categorization instead of the conventional grouping of elementary matrices and then reconstruct

to the time domain. This is achieved by an alternative method (Meth.4) that performed by using transformed features extracted from PCs with SSA. The classification method is evaluated on the artificial, adopted database and validated on professionally worked samples. Only one classifier is trained and tested. Subsequently, the nature of the frame contents can be determined by voting to for the class with the highest probability; otherwise, if the probability is between 25% and 75%, then the frame will be classified as a mix of speech and music. The study recommended that the size of the frame with this method should be longer than 90 ms, while the corresponding SSA window length should be between $\frac{Frame\_size}{4} \geq L_w \leq \frac{Frame\_size}{2}$ to achieve satisfactory results. Computer simulations present that SSA-based transformed feature extraction method performs better than any other conventional classification methods. The RFs classifier with the final method shows good and authoritative prediction sensitivities of over 89.5% with a reasonable number of features. Comparison of classification performance, specificity, and sensitivity with other baseline methods is undertaken, and its efficacy demonstrated. The performance demonstrates and verifies that the suggested approach is a promising method for overlapped recognition of mixed soundtracks. However, when the professionally mixed soundtracks were used for validation, the results were slightly lower, which is normal since the master mixing styles are used with a different dynamic range and equalisation. As mentioned in the preceding chapter, theoretically, the (Meth.4) should present the most improvements in soundtrack classification, since each PC corresponds to a particular oscillation and classified separately, which means the overlapping is almost removed.

To conclude, for overlapped soundtrack with speech and music some cleaning algorithm with the existing classification method is shown to improve the ability to detect the

overlap as apparent in Chapter 7. Augmented features over a slightly longer period can further improve the ability to handle the overlapped audio. Ultimately, an algorithm such as SSA has shown the best result evidenced by Figure 10-5 in handling overlapped between speech and music, since SSA internally separate these components.

The major limitation is that the used audio mixture based upon the music and speech. Hence, no event sound or ambient noise were involved. Therefore, it is hard to conclude if these methods will still be efficient when the mixture involve an event sound. However, the internal feature separation mechanism is a single channel component decomposition. The method itself does not limit the number of the included components, to be separated. Although the suitable grouping criterion and techniques need to be identified for the additional event components, there is no foreseeable reason why it should not work.

Finally, all objectives documented were positively met. SSA can be considered an effective tool for complete separation of overlapped soundtracks with reference to speech and music for further information retrieval. This produces that the suggested two algorithms are highly useful methods for overlapped audio classification.

11.2 Future Work

Despite the fact that the results demonstrated in this study have shown the efficiency of the SSA for decomposition of mixed soundtracks and transformed feature extraction, further validation work could supplement this and the research could go further in a number of ways:

- Expanding the proposed method for the event sound class: Event sound recognition is challenging since there is often no prior knowledge as to how many different types of the event there are likely to be, and which ones are of interest. Hence, an ad-

hoc approach might be best for event processing based on the application, due to event sounds being an open set. This is included the recording and collective of samples.

- Investigation of the effect of ambient noise on classification performance.

- Extending the method for real-time soundtracks classification: The overlapping sliding windows from one of the essential requirements for real-time processing, as deployed in this study. Another requirement is increased memory and computational load for the system. Furthermore, the ambient noise problem should be considered, as has been discussed the scope is limited to the microphone noise in this research.

- Examining the feature space by sensitivity with reference to the change in mixing ratio of speech and music. This could be useful in developing a regression model for predicting a mixing proportion between the mixed classes.

- Extending the method for speech and music separation.

- Investigating other machine learning techniques with the proposed method such as NNT and Deep Learning Techniques. There is no classifier, which can be considered ideal; the best one's performance is that which correctly classifies unseen cases with generality. Perfect performance of the proposed system is related to both module training design and testing. Infrequently, complicated classifiers present the over-fitting problem, fitting predictions to the training data and achieving poor performance when applied with unseen samples. Expanding to more than one classifier and deploying different techniques can provide improved classification accuracy and consistency.

REFERENCES

Aksoy, S. and Haralick, R.M., 2001. Feature normalization and likelihood-based similarity measures for image retrieval. Pattern recognition letters, 22(5), pp.563-582.

Alexandrov, T., Golyandina, N., and Spirov, A. (2008). Singular spectrum analysis of gene expression profiles of early Drosophila embryo: exponential-in-distance patterns. Research letters in signal processing, 2008, 12.

AL-Maathidi, M. and Li, F. F. Feature spaces and machine learning regime for audio content classification and indexing. The International Conference on Computing, Networking and Digital Technologies (ICCNDT2012), 2012. The Society of Digital Information and Wireless Communication, 335-347.

Arriola, Y. and Carrasco, R. 1990a. Integration of multi-layer perceptron and Markov models for automatic speech recognition. UK IT 1990 Conference. United Kingdom: IET.

Arriola, Y. and CARRASCO, R. A. 1990b. Parallel algorithms for automatic speech recognition. Techniques for Speech Processing, IEE Colloquium on.

Aubé, W., Angulo-Perkins, A., Peretz, I., Concha, L. and Armony, J.L., 2014. Fear across the senses: Brain responses to music, vocalizations and facial expressions. Social cognitive and affective neuroscience, p.nsu067.

Bachu, R., Kopparthi, S., Adapa, B. and Barkana, B., 2008, Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal. American Society for Engineering Education (ASEE) Zone Conference Proceedings. pp. 1-7.

BAE, C., CHUNG, Y. Y., SHUKRAN, M. A. M., CHOI, E. and YEH, W.-C. An intelligent classification algorithm for LifeLog multimedia applications. Multimedia Signal Processing, 2008 IEEE 10th Workshop on, 2008. IEEE, pp. 558-562.

Barbedo, J.G.A. and Tzanetakis, G., 2010, March. Instrument identification in polyphonic music signals based on individual partials. In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, pp. 401-404, IEEE.

Bartkiewicz P., 2013. “CLAM (C++ Library for Audio and Music)", [online] Available at: http://clam-project.org/index.html/. [2014].

Berger, A. L., Pietra, V. J. D. and Pietra, S. A. D. 1996. A maximum entropy approach to natural language processing. Computational linguistics, 22, 39-71.

BHARUCHA, J. and KRUMHANSL, C. L. 1983. The representation of harmonic structure in music: Hierarchies of stability as a function of context. Cognition, 13, 63-102.

Bishop, C. M. 2006. Pattern Recognition and Machine Learning.

Blackman, R. B. and Tukey, J. W. (1959), The Measurement of Power Spectra from the Point of View of Communications Engineering — Part I. Bell System Technical Journal, 37: 185–282. doi:10.1002/j.1538-7305.1958.tb03874.x

Boakye, K., Trueba-hornero, B., Vinyals, O. and FRIEDLAND. , 2008, G. Overlapped speech detection for improved speaker diarization in multiparty meetings. 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4353-4356.

Bogdanov, D., Wack N., Gómez E., Gulati S., Herrera P., Mayor O., et al. 2013. “Essentia ". [online] Available at: http://essentia.upf.edu/. [April. 2014].

Bonizzi, P., Karel, J.H.M., Zeemering, S. and Peeters, R.L.M., 2015, September. Sleep apnea detection directly from unprocessed ECG through singular spectrum decomposition. In 2015 Computing in Cardiology Conference (CinC), pp. 309-312. IEEE.

Breiman, L. 1996a. Bagging predictors. Machine Learning, 24, 123-140.

Breiman, L. 2001a. Random Forests. Machine Learning, 45, pp. 5-32.

Breiman, L. 2001b. Statistical modelling: The two cultures. Statistical Science, 16, 199–231.

Breiman, L., 1996b. Out-of-bag estimation (pp. 1-13). Technical report, Statistics Department, University of California Berkeley, Berkeley CA 94708, 1996b. 33, 34

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and

regression trees. New York: Chapman & Hall.

Briefer, E., Osiejuk, T.S., Rybak, F. and Aubin, T., 2010. Are bird song complexity and song sharing shaped by habitat structure? An information theory and statistical approach. Journal of Theoretical Biology, 262(1), pp.151-164.

Broomhead, D. S., and King, G. P. (1986). Extracting qualitative dynamics from experimental data. Physica D: Nonlinear Phenomena, 20(2), 217-236. doi: http://dx.doi.org/10.1016/0167-2789(86)90031-X

Cannam C., 2010. "MIRtoolbox Open Source", Queen Mary, University of London [online] Available at: http://sonicvisualiser.org/. [2014].

CHEN, C.-H. 1988. Signal processing handbook, CRC Press.

Chou, W. and Gu, L., 2001. Robust singing detection in speech/music discriminator design. In Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on (Vol. 2, pp. 865-868).

Chris, F. 1986. "Nyquist". [online] Available at: https://sourceforge.net/projects/nyquist/ [1 Apr. 2014].

Claessen, D. and Groth, D., 2002. A beginner's guide to SSA. CERES-ERTI, Ecole Normale Superieure, Paris. COMON, P. and JUTTEN, C. 2010. Handbook of Blind Source Separation: Independent component analysis and applications, Academic press.

Collins, T., Tillmann, B., Barrett, F.S., Delbe, C. and Janata, P., 2014. A combined model of sensory and cognitive representations underlying tonal expectations in music: from audio signals to behavior. Psychological review, 121(1), p.33.

Comon, P. and Jutten, C. eds., 2010. Handbook of Blind Source Separation: Independent component analysis and applications. Academic press.

Comon, P., 1994. Independent component analysis, a new concept?. Signal processing_ Elsevier, 36(3), pp.287-314.

Davis, S. and Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE transactions on acoustics, speech, and signal processing, 28(4), pp.357-366.

Dhanalakshmi, P., Palanivel, S. and Ramalingam, V., 2009. Classification of audio signals using SVM and RBFNN. Expert systems with applications, 36(3), pp.6069-6075.

Díaz-Uriarte, R. and De Andres, S.A., 2006. Gene selection and classification of microarray data using random forest. BMC bioinformatics, 7(1), p.1.

Dietterich, T.G., 2000. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine learning, 40(2), pp.139-157.

Dixon S. 2001. "BeatRoot". [online] Available at: https://code.soundsoftware.ac.uk/projects/beatroot. [11 March. 2014].

Downie, J. S. 7 April 2007. The International Music Information Retrieval Systems Evaluation Laboratory (IMIRSEL) Project [Online]. Available: http://music-ir.org/evaluation/ [Accessed 10 March/2014 2014].

Downie, J.S., 2003, October. Toward the scientific evaluation of music information retrieval systems. In ISMIR.

Duncan, P.J., Mohammed, D. and Li, F.F., 2015, July. Audio information extraction from arbitrary sound recordings. 22nd International Congress on Sound and Vibration (ICSV22) Conference Proceedings.

Duncan, P.J., Mohammed, D.Y. and Li, F.F., 2014, April. Audio Information Mining–Pragmatic Review, Outlook, and a Universal Open Architecture. In Audio Engineering Society Convention 136. Audio Engineering Society.

Eftaxias, K., Enshaeifar, S., Geman, O., Kouchaki, S. and Sanei, S., 2015, July. Detection of Parkinson's tremor from EMG signals; a singular spectrum analysis approach. In Digital Signal Processing (DSP), IEEE International Conference on (pp. 398-402).

El-Maleh, K., Klein, M., Petrucci, G. and Kabal, P., 2000. Speech/music discrimination for multimedia applications. In Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on (Vol. 6, pp. 2445-2448). IEEE.

Elsner, J. B. and Tsonis, A. A. 2013. Singular spectrum analysis: a new tool in time series analysis, Springer Science and Business Media.

Enshaeifar, S., Kouchaki, S., Took, C.C. and Sanei, S., 2016. Quaternion Singular Spectrum Analysis of Electroencephalogram With Application in Sleep Analysis. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 24(1), pp.57-67.

Eyben, F. 2016. Real-time speech and music classification by large audio feature space extraction, Springer

Fleischer, H. 1976. Über die Wahrnehmbarkeit von Phasenänderungen. Acustica (im Druck, 1976b).

Fukunaga, K. 1970. Introduction to Statistical Pattern Recognition, New York, Academic Press.

G Nike Gnanateja, P. M. 2012. Mixing the speech signals and noise at desired Signal to Noise Ratios [Online]. India Institute of Speech and hearing. Available:https://uk.mathworks.com/matlabcentral/fileexchange/view_license?file_info_id=37842.

Ghaderi, F., Mohseni, H.R. and Sanei, S., 2011. Localizing heart sounds in respiratory signals using singular spectrum analysis. IEEE Transactions on Biomedical Engineering, 58(12), pp.3360-3367.

Ghil, M., Allen, M.R., Dettinger, M.D., Ide, K., Kondrashov, D., Mann, M.E., Robertson, A.W., Saunders, A., Tian, Y., Varadi, F. and Yiou, P., 2002. Advanced spectral methods for climatic time series. Reviews of geophysics, 40(1).

Giannakopoulos, T. 2009. Study and application of acoustic information for the detection of harmful content, and fusion with visual information. Department of Informatics and Telecommunications, vol. PhD. University of Athens, Greece.

Giannakopoulos, T. and Pikrakis, A., 2014. Introduction to Audio Analysis: A MATLAB® Approach. Academic Press, Elsevier.

Giannoulis, D., Benetos, E., Stowell, D., Rossignol, M., Lagrange, M. and Plumbley, M.D., 2013, October. Detection and classification of acoustic scenes and events:

an IEEE AASP challenge. In 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (pp. 1-4). IEEE.

GOLD, B., MORGAN, N. AND ELLIS, D. 2011. Speech and audio signal processing: processing and perception of speech and music, Canada, John Wiley and Sons.

Golyandina, N. and Zhigljavsky, A., 2013. Singular Spectrum Analysis for time series. Springer Science and Business Media.

Golyandina, N., Nekrutkin V., and Kirill B., "Time Series Analysis And Forecasting, Caterpillar SSA Method". Gistatgroup.com. N.p., 2017. Web. 13 Aug. 2016.

Golyandina, N., Nekrutkin, V. and Zhigljavsky, A.A., 2002. Analysis of time series structure: SSA and related techniques (chapman & hall crc monographs on statistics & applied probability).

Golyandina, N.E. and Lomtev, M.A., 2016. Improvement of separability of time series in singular spectrum analysis using the method of independent component analysis. Vestnik St. Petersburg University: Mathematics, 49(1), pp.9-17.

Harrington, P., 2012. Machine learning in action (Vol. 5). Greenwich, CT: Manning.

Harris, F.J., 1978. On the use of windows for harmonic analysis with the discrete Fourier transform. Proceedings of the IEEE, 66(1), pp.51-83.

Harris, T.J. and Yuan, H., 2010. Filtering and frequency interpretations of singular spectrum analysis. Physica D: Nonlinear Phenomena, 239(20), pp.1958-1967.

Hassani, H. 2010. A brief introduction to singular spectrum analysis. Optimal decisions in statistics and data analysis

Hastie, T., Tibshirani, R., Friedman, J. and Franklin, J. 2005. The elements of statistical learning: data mining, inference and prediction, the Second edition California, springer.

Heittola, T., Mesaros, A., Virtanen, T. and Eronen, A., 2011. Sound event detection in multisource environments using source separation. Proc CHiME, pp.36-40.

Hérault, J., Jutten, C. and Ans, B., 1985. Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In 10 Colloque sur le traitement du signal et des images,

FRA, 1985. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images.

Huang, R. and Hansen, J.H., 2006. Advances in unsupervised audio classification and segmentation for the broadcast news and NGSW corpora. IEEE Transactions on audio, speech, and language processing, 14(3), pp.907-919.

Huang, Y.-F., Lin, S.-M., Wu, H.-Y. and LI, Y.-S. 2014. Editorial: Music genre classification based on local feature selection using a self-adaptive harmony search algorithm. Data Knowl. Eng., 92, 60-76.

Huron D., 1995. "The Humdrum Toolkit: Software for Music Research". [online] Available at: http://www.humdrum.org/. [April. 2014].

Jarchi, D. and Yang, G.Z., 2013, May. Singular spectrum analysis for gait patterns. In 2013 IEEE International Conference on Body Sensor Networks (pp. 1-6). IEEE.

Jensen, K., 1999. *Timbre models of musical sounds*. Department of Computer Science, University of Copenhagen.

Jiang, X., Zhang, T., Hu, X., Lu, L., Han, J., Guo, L. and Liu, T., 2012, October. Music/speech classification using high-level features derived from fMRI brain imaging. In Proceedings of the 20th ACM international conference on Multimedia (pp. 825-828). ACM.

Joder, C., Weninger, F., Eyben, F., Virette, D. and Schuller, B., 2012, March. Real-time speech separation by semi-supervised nonnegative matrix factorization. In International Conference on Latent Variable Analysis and Signal Separation (pp. 322-329). Springer Berlin Heidelberg.

Juslin, P.N., 2000. Cue utilization in communication of emotion in music performance: Relating performance to perception. Journal of Experimental Psychology: Human perception and performance, 26(6), p.1797.

Kargupta, K., Park, B.H. and Dutta, H., 2006. Orthogonal decision trees. IEEE Transactions on Knowledge and Data Engineering, 18(8), pp.1028-1042.

Kaur1, S., Garg2, A. and Bactor3, P. 2013. An overview of speech enhancement techniques and evaluation. International journal for advance research in engineering and technology, 1, 27-30.

Kenny, J. and Keeping, E. 1962. Root Mean Square. Mathematics of Statistics, Pt, 1, 59-60.

Khaldi, K., Boudraa, A.O. and Turki, M., 2016. Voiced/unvoiced speech classification-based adaptive filtering of decomposed empirical modes for speech enhancement. IET Signal Processing, 10(1), pp.69-80.

Khemiri, H., Chollet, G. and Petrovska-Delacrétaz, D. 2013. Automatic detection of known advertisements in radio broadcast with data-driven ALISP transcriptions. Multimedia Tools and Applications, 62, 35-49

Khonglah, B.K. and Prasanna, S.M., 2016. Speech/music classification using speech-specific features. Digital Signal Processing, 48, pp.71-83.

Kim, H.G., Moreau, N. and Sikora, T. 2005. MPEG-7 Audio and Beyond Audio Content Indexing or Retrieval

Knox, D., Beveridge, S., Mitchell, L.A. and MacDonald, R.A., 2011. Acoustic analysis and mood classification of pain-relieving music. The Journal of the Acoustical Society of America, 130(3), pp.1673-1682.

Kohavi, R., 1995, August. A study of cross-validation and bootstrap for accuracy estimation and model selection. In Ijcai (Vol. 14, No. 2, pp. 1137-1145)

Kos, M., Kačič, Z. and Vlaj, D., 2013. Acoustic classification and segmentation using modified spectral roll-off and variance-based features. Digital Signal Processing, 23(2), pp.659-674

Lartillot O., 2010. “MIRtoolbox Open Source". [online] Available at: http://mirtoolbox.sourceforge.net/. [2014].

Lartillot, O., Toiviainen, P. and Eerola, T., 2008. A matlab toolbox for music information retrieval. In Data analysis, machine learning and applications (pp. 261-268). Springer Berlin Heidelberg.

Laukka, P., Juslin, P. and Bresin, R., 2005. A dimensional approach to vocal expression of emotion. Cognition & Emotion, 19(5), pp.633-653.

Lavner, Y. and Ruinskiy, D., 2009. A Decision-Tree-Based Algorithm for Speech/Music Classification and Segmentation. EURASIP Journal on Audio,

Speech, and Music Processing, 2009, 1-14.

Lee, D.D. and Seung, H.S., 2001. Algorithms for non-negative matrix factorization. In Advances in neural information processing systems (pp. 553-562).

Lee, K. and Ellis, D.P., 2008, March. Detecting music in ambient audio by long-window autocorrelation. In 2008 IEEE International Conference on Acoustics, Speech and Signal Processing (p. 9-12).

LEE, T. K. M., GAN, S. S. W., Sanei, S. and Kouchaki, S., 2013.Assessing rehabilitative reach and grasp movements with Singular Spectrum Analysis. 21st European Signal Processing Conference (EUSIPCO 2013), 9-13 Sept. 1-5.

Lefèvre, S. and Vincent, N., 2011, A two level strategy for audio segmentation. Digital Signal Processing, 21, 270-277.

Lika, R.A., Seldon, H.L. and Kiong, L.C., 2014, August. Feature analysis of speech emotion data on arousal-valence dimension using adaptive neuro-fuzzy classifier. In Industrial Automation, Information and Communications Technology (IAICT), 2014 International Conference on (pp. 104-110). IEEE.

Liu, M. and Wan, C., 2001. A study on content-based classification and retrieval of audio database. In *Database Engineering and Applications, 2001 International Symposium on.* (pp. 339-345). IEEE.

Liu, Y. H., Zhou, D. M. and Jiang, Z. J., 2013, An improved spectral subtraction speech enhancement algorithm under non-stationary noise. Advanced Materials Research. Trans Tech Publ, 533-541.

Loizou, P. C. 2013. Speech enhancement: theory and practice, CRC press.

Lu, L., Jiang, H. and Zhang, H., 2001a, October. A robust audio classification and segmentation method. In Proceedings of the ninth ACM international conference on Multimedia (pp. 203-211). ACM.

Lu, L., Li, S.Z. and Zhang, H.J., 2001b, August. Content-based audio segmentation using support vector machines. In Proc. ICME (Vol. 1, pp. 749-752).

Lu, L., Zhang, H.J. and Jiang, H., 2002. Content analysis for audio classification and segmentation. IEEE Transactions on speech and audio processing, 10(7), pp.503-

516.

Lu, Y. and Saniie, J., 2015, October. Singular spectrum analysis for trend extraction in ultrasonic backscattered echoes. In Ultrasonics Symposium (IUS), 2015 IEEE International (pp. 1-4).

Lu, Y.C., Wu, C.-W., Lu, C.-T. and Lerch, A., 2016, An Unsupervised Approach to Anomaly Detection in Music Datasets. Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. Pisa, Italy: ACM.

Luellen, J.K., Shadish, W.R. and Clark, M.H., 2005. Propensity scores an introduction and experimental test. Evaluation Review, 29(6), pp.530-558.

Ma, H.G., Lei, R., Kong, X.Y., Liu, Z.Q. and Jiang, Q.B., 2012, October. Determine a proper window length for singular spectrum analysis. In Radar Systems (Radar 2012), IET International Conference on (pp. 1-6). IET.

Macchi, O. and Moreau, E., 1997. Self-adaptive source separation. I. Convergence analysis of a direct linear network controlled by the Herault-Jutten algorithm. IEEE Transactions on Signal Processing, 45(4), pp.918-926.

Maimon, O. and Rokach, L. 2008. Data mining with decision trees: theory and applications [Online]. USA: World Scientific Publishing.

Mamou, J. and Feleppa, E. J. 2007. Singular spectrum analysis applied to ultrasonic detection and imaging of brachytherapy seeds. The Journal of the Acoustical Society of America, 121, 1790-1801.

McCartney J. 1996. “SuperCollider". [online] Available at: http://supercollider.github.io/. [6 Apr. 2014].

McKay, C., 2009. “jMIR". [online] Available at: https://sourceforge.net/projects/jmir/. [2014].

McKay, C., 2010. Automatic music classification with jMIR (Doctoral dissertation, McGill University).

Mckinney, D. 2008. Improve Your Recordings and Mixes, on the Cheap [Online]. Available at: http://www.hometracked.com/ [Accessed 13 August 2014].

McKinney, M.F. and Breebaart, J., 2003, October. Features for audio and music classification. In ISMIR (Vol. 3, pp. 151-158).

Meinedo, H. and Neto, J., 2003, April. Audio segmentation, classification and clustering in a broadcast news task. In Acoustics, Speech, and Signal Processing, 2003. Proceedings (ICASSP'03). 2003 IEEE International Conference on (Vol. 2, pp. II-5). IEEE.

Mert, C. and MILNIKOV, A. 2011.Singular Spectrum Analysis Method as a universal filter. Application of Information and Communication Technologies (AICT), 2011 5th International Conference on, IEEE, 1-5.

Michalevsky, Y., Boneh, D. and Nakibly, G., 2014, August. Gyrophone: Recognizing Speech from Gyroscope Signals. In USENIX Security (pp. 1053-1067).

Microsoft and Cambridge University. Sept. 28, 2000. Speech Recognition Toolkit [Online]. Available: https://www.microsoft.com/enus/news/press/2000/sept00/cambridge-pr.aspx [Accessed 28/12/2013 2013].

Misra, H., IKBAL, S., BOURLARD, H. and HERMANSKY, H. 2004. Spectral entropy based feature for robust ASR. Acoustics, Speech, and Signal Processing, 2004. Proceedings (ICASSP'04). IEEE International Conference on, IEEE, vol. 1.193-6.

Mitchell, T. 1997. Machine learning, New Delhi, India, McGraw Hill Education

Mitrovic, D., Zeppelzauer, M. and Breiteneder, C. 2010. Features for Content-Based Audio Retrieval. In: MARVIN, V. Z. (ed.) Advances in Computers. Elsevier

Miyazaki, R., Saruwatari, H., Inoue, T., Takahashi, Y., Shikano, K. and Kondo, K. 2012. Musical-Noise-Free Speech Enhancement Based on Optimized Iterative Spectral Subtraction. Audio, Speech, and Language Processing, IEEE Transactions on, 20, 2080-2094

Mohammadi, S.M., Kouchaki, S., Ghavami, M. and Sanei, S., 2016. Improving time–frequency domain sleep EEG classification via singular spectrum analysis. *Journal of Neuroscience Methods*, *273*, pp.96-106.

Mohammed, D. Y., Duncan, P. J., AL-Maathidi, M. M. and LI, F. F. 2015. A system for semantic information extraction from mixed soundtracks deploying MARSYAS framework. Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on Industrial Informatics INDIN Cambridge, UK: IEEE.

Mohri, M., Moreno, P.J. and Weinstein, E., 2007, September. Robust Music Identification, Detection, and Analysis. In ISMIR (pp. 135-138).

Noyes, J. and Starr, A. 1996. Use of automatic speech recognition: current and potential applications. Computing and Control Engineering Journal, 7, 203-208.

Palo, H. K., Mohanty, M. N. and Chandra, M. 2015. New features for emotional speech recognition. 2015 IEEE Power, Communication and Information Technology Conference (PCITC), 15-17 Oct. 2015. 423-429.

Panagiotakis, C. and Tziritas, G. 2005. A speech/music discriminator based on RMS and zero-crossings. Multimedia, IEEE Transactions on, 7, 155-166.

PAUL BOERSMA, D. W. Praat: Doing phonetics by computer [Online]. The Netherlands. Available: http://www.fon.hum.uva.nl/praat/download_win.html 2015].

Pearson, K., 1901. On Lines and Planes of Closest Fit to System of Points in Space. Philosophical Magazine, 2, 559-572.

Penland, C., Ghil, M. and Weickmann, K. M. 1991. Adaptive filtering and maximum entropy spectra with application to changes in atmospheric angular momentum. Journal of Geophysical Research: Atmospheres, 96, 22659-22671.

Peruse G., 2015. “ChucK: Strongly-timed, Concurrent, and On-the-fly music Programming Language”, [online] Available at: http://chuck.cs.princeton.edu/. [2016].

Plomp, R. and LEVELT, W. J. M. 1965. Tonal consonance and critical bandwidth. The journal of the Acoustical Society of America, 38, 548-560.

Quinlan, J. R. 1986. Induction of decision trees. Machine learning, 1, 81-106.

Rifkin, R. and Klautau, A. 2004. In defense of one-vs-all classification. Journal of machine learning research, 5, 101-141.

Rokach, L. and Maimon, O., 2014. Data mining with decision trees: theory and applications. World scientific.

Ruelle, D. (1984). Strange attractors. Universality in Chaos. P. Cvitanovic, editor. Adam Hilger Ltd., Bristol, UK, 37-48.

Rukhin, A. L. (2002). Analysis of time series structure SSA and Related techniques. Techno-metrics, 44(3), 290-290.

Sanei, S. and Hassani, H., 2015. Singular spectrum analysis of biomedical signals. CRC Press.

Sanei, S. and Hosseini-Yazdi, A. (2011, 3-8 July). Extraction of ECG from single channel EMG signal using constrained singular spectrum analysis. Paper presented at the International Conference on Digital Signal Processing (DSP).

Sanei, S., GHODSI, M. and HASSANI, H., 2011b. An adaptive singular spectrum analysis approach to murmur detection from heart sounds. Medical Engineering and Physics, 33, 362-367.

SATTAR, F., DRIESSEN, P. F., TZANETAKIS, G. and PAGE, W. H. 2011, A new method for classification of events in noisy hydrophone data. Communications, Computers and Signal Processing (PacRim), 2011 IEEE Pacific Rim Conference on, 23-26 Aug. 2011. 660-667.

Saunders, J., 1996. Real-time discrimination of broadcast speech/music. Acoustics, Speech, and Signal ProcessingICASSP-96. Conference Proceedings. 1996 IEEE International Conference on, 7-10 May 1996. 993-996 vol. 2.

Scheirer, E. and Slaney, M. 1997. Construction and evaluation of a robust multifeature speech/music discriminator. Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on, 21-24 Apr 1997. Vol.2. 1331-1334.

Schottstaedt, B. 1986. "CLM". https://ccrma.stanford.edu/software/snd/snd/clm.html., Mar. 2015.

Schuller, B. W. 2013. Intelligent Audio Analysis, Berlin, Springer Berlin Heidelberg.

Sebastian, S. and Rathnakara, S., 2013. Separation and localisation of heart sound

artefacts from respiratory data by adaptive selection of Eigen triples in singular spectrum analysis. Computing, Communications and Networking Technologies (ICCCNT), Fourth International Conference, pp. 1-5.

Segal, M.R., 2004. Machine learning benchmarks and random forest regression. Centre for Bioinformatics & Molecular Biostatistics.

SETHARES, W. 1998. Tuning, timbre, spectrum, scale Springer. NY.

Seyerlehner, K., Pohle, T., Schedl, M. and Widmer, G., 2007, September. Automatic music detection in television productions. In Proceedings of the 10th International Conference on Digital Audio Effects (DAFx'07).

Shannon, C. E. 1948. A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review, 5, 3-55.

Shao, X., Xu, C. and Kankanhalli, M.S., 2003, December. Applying neural network on the content-based audio classification. In Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on (Vol. 3, pp. 1821-1825). IEEE.

Shete, D. and Patil, S. 2014. Zero crossing rate and Energy of the Speech Signal of Devanagari Script. IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), Volume 4, PP 01-05.

Shokouhi, N., Ziaei, A., Sangwan, A. and Hansen, J.H., 2015, April. Robust overlapped speech detection and its application in word-count estimation for Prof-Life-Log data. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4723-4728). IEEE.

Smaragdis, P., Raj, B. and Shashanka, M., 2007, September. Supervised and semi-supervised separation of sounds from single-channel mixtures. In International Conference on Independent Component Analysis and Signal Separation (pp. 413-421). Springer Berlin Heidelberg.

Statnikov, A., Wang, L. and Aliferis, C.F., 2008. A comprehensive comparison of

random forests and support vector machines for microarray-based cancer classification. BMC bioinformatics, 9(1), p.1.

Stewart, W.J., 2009. Probability, Markov chains, queues, and simulation: the mathematical basis of performance modeling. Princeton University Press.

Strobl, C., Malley, J. and Tutz, G., 2009. An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests. Psychological methods, 14(4), p.323.

STURM, B. L. An analysis of the GTZAN music genre dataset. Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies, 2012. ACM, 7-12.

Temko, A. and Nadeu, A. C. 2007. Acoustic Event Detection and Classification. PhD Thesis.

Temko, A. and Nadeu, C. 2006a. Classification of acoustic events using SVM-based clustering schemes. Pattern Recognition, 39, 682-694.

Temko, A., Malkin, R., Zieger, C., Macho, D., Nadeu, C. and Omologo, M., 2006b, Acoustic event detection and classification in smart-room environments: Evaluation of CHIL project systems. Cough, 65(48), p.5.

Thambi, S.V., Sreekumar, K.T., Kumar, C.S. and Raj, P.R., 2014, December. Random forest algorithm for improving the performance of speech/non-speech detection. In Computational Systems and Communications (ICCSC), 2014 First International Conference on (pp. 28-32). IEEE.

Toh, A.M., Togneri, R. and Nordholm, S., 2005. Spectral entropy as speech features for speech recognition. Proceedings of PEECS, 1.

Tomonori, I., Ryo, M. and Kunio, K., 2008, A background music detection method based on robust feature extraction. IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 13-16.

Tzanetakis G., 2009. “MARSYAS", [online] Available at: http://marsyas.info/ . [2014].

Tzanetakis, G. 2002. Manipulation, Analysis and Retrieval Systems for Audio

Signals. doctor of philosophy, facultyof princeton university in Candidacy.

Tzanetakis, G. 2005 Audio-based gender identification using bootstrapping. Communications, Computers and signal Processing, 2005. PACRIM. 2005 IEEE Pacific Rim Conference on, 23-26 Aug. 432-433.

Tzanetakis, G. 2009. Marsyas (Music Analysis, Retrieval and Synthesis for Audio Signals) [Online]. Available: http://marsyas.info/ [Accessed 01 March 2014].

Tzanetakis, G. and Cook, P. 1999a. MARSYAS: a framework for audio analysis. Org. Sound, 4, 169-175.

Tzanetakis, G. and Cook, P. 1999b. Multifeature audio segmentation for browsing and annotation. Applications of Signal Processing to Audio and Acoustics, 1999 IEEE Workshop on,103-106.

Tzanetakis, G. and Cook, P. 2002. Musical genre classification of audio signals. Speech and Audio Processing, IEEE Transactions on, 10, 293-302.

University Of Salford, University Of Salford Studios And Labs For Acoustic Engineering, Audio And Video Engineering | University Of Salford | Manchester. Acoustics.salford.ac.uk. Web. 23 Oct. 2016.

UNIVERSITY, M. A. C. Sept. 28, 2000. Speech Recognition Toolkit [Online]. Available: https://www.microsoft.com/en-us/news/press/2000/sept00/cam-bridgepr.aspx [Accessed 28/12/2013 2013].

Upadhyay, n. and Karmakar, A. 2015. Speech Enhancement using Spectral Subtraction-type Algorithms: A Comparison and Simulation Study. Procedia Computer Science, 54, 573-584.

Upadhyay, N. and Karmakar, A., 2012, December. Single channel speech enhancement utilizing iterative processing of multi-band spectral subtraction algorithm. In Power, Control and Embedded Systems (ICPCES), 2012 2nd International Conference on (pp. 1-6). IEEE.

Vautard, R. and Ghil, M., 1989. Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series. Physica D: Nonlinear Phenomena, 35(3), pp.395-424.

Vautard, R., Yiou, P. and Ghil, M. 1992. Singular-spectrum analysis: A toolkit for short, noisy chaotic signals. Physica D: Nonlinear Phenomena, 58, 95-126.

Waibel1, A., Steusloff2, H., Stiefelhagen1, R. and Consortium, T. C. P. 2004. CHIL: COMPUTERS IN THE HUMAN INTERACTION LOOP. '1Interactive Systems Labs (ISL), University of Karlsruhe Fraunhofer Institut für Informations- und Datenverarbeitung (IITB)'.

Wang, N.-C., Hudson, R. E., Tan, L. N., Taylor, C. E., Alwan, A. and Yao, K. 2013. Bird phrase segmentation by entropy-driven change point detection. Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, 773-777.

Wang, Y., Liu, Z. and Huang, J.-C. 2000. Multimedia content analysis-using both audio and visual clues. Signal Processing Magazine, IEEE, 17, 12-36.

Wang, Y., Liu, Z., and Dong, B. (2016). Heart rate monitoring from wrist-type PPG based on singular spectrum analysis with motion decision. Paper presented at the 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).

Webb, A. R. 2011. Statistical pattern recognition, John Wiley and Sons.

Weiss, M., Aschkenasy, E. and Parsons, T. 1975. Study and development of the INTEL technique for improving speech intelligibility. DTIC Document.

Weng, Z., Li, L. and Guo, D., 2010, July. Speaker recognition using weighted dynamic MFCC based on GMM. In 2010 International Conference on Anti-Counterfeiting, Security and Identification (pp. 285-288). IEEE.

Weninger, F., Geiger, J., Wöllmer, M., Schuller, B. and Rigoll, G., 2011, September. The Munich 2011 CHiME challenge contribution: NMF-BLSTM speech enhancement and recognition for reverberated multisource environments. In Proc. Machine Listening in Multisource Environments (CHiME 2011), satellite workshop of Interspeech (pp. 23-29).

Wilson, K.W., Raj, B., Smaragdis, P. and Divakaran, A., 2008, March. Speech denoising using nonnegative matrix factorization with priors. In Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference

on (pp. 4029-4032). IEEE.

WOLD, E., BLUM, T., KEISLAR, D. and WHEATEN, J. 1996. Content-based classification, search, and retrieval of audio. IEEE multimedia, 3, 27-36.

WYSE, L. and SMOLIAR, S. 1995. Toward content-based audio indexing and retrieval and a new speaker discrimination technique. Proc. ICJAI, 95.

Zeng, T., Ma, J. and Dong, M., 2014, Environmental noise elimination of heart sound based on singular spectrum analysis. Biomedical Engineering Conference (CIBEC), Cairo International. IEEE, 158-161.

ZHANG, P., ZHENG, X., ZHANG, W., LI, S., QIAN, S., HE, W., ZHANG, S. and WANG, Z. 2015. A Deep Neural Network for Modeling Music. Proceedings of the 5th ACM on International Conference on Multimedia Retrieval. Shanghai, China: ACM.

Zhang, T. and KUO, C. C. J. 2001. Audio content analysis for online audio-visual data segmentation and classification. IEEE Transactions on Speech and Audio Processing, 9, 441-457.

Zhang, Y. and LV, D.J., 2015. Selected Features for Classifying Environmental Audio Data with Random Forest. Open Automation and Control Systems Journal, 7, pp.135-142.

APPENDIX A: TABLES

Table I MCR of Speech/Mix against Music T-test Statistics for mfcc coefficients

	Target	N	Mean	Std.	Std. Error	Levene's Test for Equality of Variances	
						F	Sig.
MFCC1	Mix/Speech	13171	.3882	.15489	.00135	99.619	.000
	Music	13145	.3525	.16396	.00143		
MFCC2	Mix/Speech	13171	.4005	.15447	.00135	56.290	.000
	Music	13145	.3629	.15814	.00138		
MFCC3	Mix/Speech	13171	.3911	.15210	.00133	37.394	.000
	Music	13145	.3810	.15833	.00138		
MFCC4	Mix/Speech	13171	.3958	.15101	.00132	58.858	.000
	Music	13145	.3891	.15965	.00139		
MFCC5	Mix/Speech	13171	.4046	.15468	.00135	8.524	.004
	Music	13145	.3944	.15753	.00137		
MFCC6	Mix/Speech	13171	.4091	.15303	.00133	12.671	.000
	Music	13145	.3960	.15967	.00139		
MFCC7	Mix/Speech	13171	.4124	.15427	.00134	.082	.774
	Music	13145	.4002	.15752	.00137		
MFCC8	Mix/Speech	13171	.4135	.15225	.00133	4.681	.030
	Music	13145	.3964	.15668	.00137		
MFCC9	Mix/Speech	13171	.4122	.15012	.00131	21.425	.000
	Music	13145	.3976	.15819	.00138		
MFCC10	Mix/Speech	13171	.4182	.15200	.00132	10.848	.001
	Music	13145	.3968	.15842	.00138		
	Music	13145	.3969	.16008	.00140		
MFCC13	Mix/Speech	13171	.4199	.15361	.00134	.026	.873
	Music	13145	.3971	.15695	.00137		
MFCC14	Mix/Speech	13171	.4184	.15234	.00133	7.045	.008
	Music	13145	.3973	.15908	.00139		
MFCC15	Mix/Speech	13171	.4193	.15249	.00133	6.422	.011
	Music	13145	.3949	.15876	.00138		
MFCC16	Mix/Speech	13171	.4180	.15074	.00131	67.876	.000
	Music	13145	.3916	.16200	.00141		
MFCC17	Mix/Speech	13171	.4170	.15218	.00133	66.079	.000
	Music	13145	.3857	.16247	.00142		
MFCC18	Mix/Speech	13171	.4169	.15347	.00134	38.705	.000
	Music	13145	.3875	.16223	.00141		
MFCC19	Mix/Speech	13171	.4187	.15363	.00134	21.090	.000
	Music	13145	.3898	.16041	.00140		
MFCC20	Mix/Speech	13171	.4212	.15434	.00134	4.589	.032
	Music	13145	.3930	.15969	.00139		
MFCC21	Mix/Speech	13171	.4203	.15251	.00133	24.452	.000
	Music	13145	.3929	.16171	.00141		
MFCC22	Mix/Speech	13171	.4207	.15168	.00132	46.889	.000
	Music	13145	.3873	.16143	.00141		

Table II MCR of Speech/Mix against Music T-test Statistics for remaining features

	Target	N	Mean	Std.	Std. Error	Levene's Test for Equality of Variances	
						F	sig
RMS	Mix/Speech	13171	.3747	.15808	.00138		
	Music	13145	.3069	.14895	.00130	128.980	.000
Roughness	Mix/Speech	13171	.3723	.15657	.00136	2.678	.102
	Music	13145	.3231	.15665	.00137		
ZCR	Mix/Speech	13171	.3622	.14949	.00130	20.127	.000
	Music	13145	.3346	.15604	.00136		
Regularity	Mix/Speech	13171	.4276	.15662	.00136	1.975	.160
	Music	13145	.3894	.16129	.00141		
SC	Mix/Speech	13171	.3863	.15284	.00133	174.857	.000
	Music	13145	.3499	.16644	.00145		
Roll-off	Mix/Speech	13171	.3859	.15728	.00137	177.784	.000
	Music	13145	.3456	.17162	.00150		
Brightness	Mix/Speech	13171	.3865	.15706	.00137	92.927	.000
	Music	13145	.3463	.16654	.00145		
MIR_Centroid	Mix/Speech	13171	.3632	.17421	.00152	.228	.633
	Music	13145	.3455	.17151	.00150		
Spread	Mix/Speech	13171	.3950	.17030	.00148	53.159	.000
	Music	13145	.3642	.17474	.00152		
Entropy	Mix/Speech	13171	.3722	.15841	.00138	13.335	.000
	Music	13145	.3269	.15493	.00135		
Skewness	Mix/Speech	13171	.3894	.16679	.00145	51.869	.000
	Music	13145	.3464	.17074	.00149		
Hchange Detection	Mix/Speech	13171	.4169	.14711	.00128	10.282	.001
	Music	13145	.4233	.15011	.00131		
Entropy_FFT	Mix/Speech	13171	.4067	.15854	.00138	61.191	.000
	Music	13145	.3620	.16530	.00144		
Entrocy	Mix/Speech	13171	.4218	.17650	.00154	.916	.339
	Music	13145	.4183	.17826	.00155		

APPENDIX B: AUDIO PRODUCTION LAB

The below CD contains on the audio files that has used for the validation of the designed method.

These audio files are produces in the one of the Salford University labe with the following characteristics:

Studio C - Mackie 8 Bus Desk. Pro-tools
Dynamics
DBX 166XL
Preamp
TLA Ebony A3
Other
Digi-design Command 8
Emu Morpheus
MOTU Midi Express
HHB CDR-800
Tascam DA20
Waldorf Blofeld
UAD 2 – Duo

Monitoring
2 x Tannoy Little Golds
2 x Genelec 1029A
1 x Genelec 7050B
TLA Ivory 2
Drawmer DS201
Behringer XR2000
FX
Yamaha SPX 990
TC Electronics M One
TC Electronics D Two


Toft Audio Designs

APPENDIX D: TOOLBOX VIEW