Analysis of Speech and Music Content for Movie Genre Classification

Mrinmoy Bhattacharjee

Analysis of Speech and Music Content for Movie Genre Classification

A

Thesis submitted

for the award of the degree of

DOCTOR OF PHILOSOPHY

By

Mrinmoy Bhattacharjee

DEPARTMENT OF ELECTRONICS AND ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

GUWAHATI - 781 039, ASSAM, INDIA

January 2023

Certificate

This is to certify that the thesis entitled "**Analysis of Speech and Music Content for Movie Genre Classification**", submitted by **Mrinmoy Bhattacharjee** (156102026), a research scholar in the *Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by him under our supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and, in our opinion, has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Prof. S. R. Mahadeva Prasanna
Professor
Dept. of Electrical Engg.
Indian Institute of Technology Dharwad
Dharwad - 580 011, Karnataka, India
Dated:
Dharwad

Dr. Prithwijit Guha
Associate Professor
Dept. of Electronics & Electrical Engg.
Indian Institute of Technology Guwahati
Guwahati - 781 039, Assam, India
Dated:
Guwahati

To

My **Parents**

for their blessings and unwavering belief in me

and my **Sister**

for her love and support

Acknowledgements

I feel incredibly blessed and grateful for being able to submit my Ph.D. thesis. I am thankful that I could put in the consistent hard work required during my Ph.D. My journey would not have been as smooth and rewarding as it has been without the support and contributions of many important people in my life. Here, I attempt to convey my sincere gratitude to all of them to the best of my abilities. With utmost humility, I also beg for forgiveness from anyone whose contribution I forget to acknowledge.

I shall start by expressing my heartfelt gratitude to my Ph.D. supervisors, Prof. S. R. Mahadeva Prasanna and Dr. Prithwijit Guha, for providing an opportunity to work under their supervision. I am obliged to them for helping me improve the depth of my conceptual understanding of the research area. Their constant guidance and feedback at every step of my Ph.D. helped me remain focused and keep progressing towards completing my thesis. Their patience with my mistakes and kind attention to my efforts helped bring the thesis to its current form. I am also immensely grateful to them for graciously providing me a laboratory to work in, computing facilities for running my experiments, financial assistance for attending conferences, and other necessary logistical support.

I sincerely thank my doctoral committee members, Prof. Samarendra Dandapat (Chairman), Prof. Rohit Sinha, and Dr. V. Vijaya Saradhi, for their immense support of my research work. Their regular and timely critiques of my work and insightful suggestions to improve my research efforts significantly impacted my thesis. I am grateful to them for making me look forward to my review seminars. I am also thankful to my TA supervisors, Nodal officers of the Visvesvaraya Ph.D. scheme, and other faculty members of the EEE department of IIT Guwahati for their contributions at different stages of my Ph.D. I am grateful to the technical officers, laboratory assistants, office clerks, and other non-technical staff of the EEE department for their friendly attitude and kind support. I thank the Ministry of Electronics and Information Technology, Government of India, for financially supporting my Ph.D. I am thankful to IIT Guwahati for providing me state-of-the-art research facilities, an intellectually stimulating environment, and arguably the most serene and beautiful campus among all Indian academic institutions.

I am fortunate to have been acquainted with a group of highly motivated and knowledgeable researchers in my department. I am thankful to my seniors, Dr. Deepak KT, Dr. Abhishek Sharma,

Dr. Rajib Sharma, Dr. Bidisha Sharma, Mr. Balaji Rao Katika, Dr. Nagaraj Adiga, Dr. Banrikshem K. Khonglah, Dr. Raghavendra Kannao, Dr. Vivek Venugopal, and Dr. Nagendra Kumar for maintaining a stimulating research environment in the lab. I am grateful for the various technical and non-technical discussions with my seniors Dr. Biswajit Deb Sharma, Dr. Rohan Kumar Das, Dr. Ramesh Kumar Bhukya, Dr. Subhashis Mandal, Dr. Akhilesh Kumar Dubey, Dr. Protima Nomo Shudro, Dr. Sishir Kalita, Dr. Himakshi Choudhury, and Mr. Mathew Francis. The time during my Ph.D. would not have been easy without the help, motivation, and support of my friends, Dr. Shikha Baghel and Ms. Moakala Tzudir. I am incredibly thankful to Shikha for letting me pick her brain whenever I got stuck in any technical aspect of my work. I am also grateful to my lab mates, Dr. Sandeep Pandey, Dr. Saswati Rabha, Mr. Shoubhik Chakraborty, Dr. Deepika Gupta, Mr. Sarfaraz Jelil, Mr. Brij Nandan Tripathi, Mr. Anik Ghosh, Mr. Deep Arya, and Mr. Srihari A.

Most of all, I am indebted to my parents for providing me with the opportunity to reach this position. Their self-less love, support, and blessings have been the main reasons for my successes. In addition, I am grateful to my little sister for having the strongest confidence in me. My family's belief and trust in my abilities gave me the conviction and motivation to face all challenges. Last but not least, I am grateful to the universe for letting a speck of stardust be part of this reality and enjoy a purposeful life.

Thank you.

Mrinmoy Bhattacharjee

Abstract

Movie viewership has increased many folds over the past decades. Developing automatic methods to analyze the ever-increasing movie content has become more necessary than ever. One of the essential tasks among various applications is the automatic identification of movie genres. Movie genre information might be helpful in underage censorship, choice-based query retrieval, and targetted publicity. This thesis aims to develop systems for automatically identifying movie genres using the audio modality. The movie audio mainly consists of speech and music signals in isolated form or as overlapping mixtures. This thesis pursues the hypothesis that the presence of speech and music signals in movies might vary according to their associated genres. Therefore, a detailed analysis of these audio signals is performed in this thesis. The significant contributions of this thesis are briefly discussed next.

The first contribution of this thesis involves the proposal of a novel Spectral Peak Tracking (SPT) method that traces the peaks in the magnitude spectrogram. The SPT approach captures distinct patterns in the spectrograms of speech and music. Furthermore, two tempo-spectral features for Speech vs. Music Classification (SMC) are proposed using the SPT method. The SMC is performed by training various binary classifiers on these proposed features. The proposed approach performs well on different standard audio datasets that include speech and music signals both in isolated and overlapped conditions. The proposed method has shown decent performance on real-world audio with continuous transitions between the two audio categories. In addition, a small Hindi movie audio dataset with seven annotated audio classes is contributed to validate the speech-music detection performance on actual movie signals.

The second contribution of this thesis is the exploration of phase information in the SMC task. Existing works have mainly used various magnitude-based features for this task. Comparatively, the phase spectrum has received lesser attention in this task. The phase component is believed to carry valuable information to aid audio class identification. The potential of three existing phase-based features is highlighted through a statistical significance test and canonical correlation analysis. The proposed SMC method with phase-based features is evaluated on multiple standard datasets. In combination with magnitude-based features, the phase-based ones consistently improve the performance over the baselines for the datasets used in experiments. Various phase and magnitude-based feature fusions also perform satisfactorily in cross-dataset generalization experiments and detection of audio

classes in noise-corrupted signals. Moreover, the combination of phase-based and magnitude-based features effectively segments continuous speech and music signal sequences.

The third contribution of this thesis deals with detecting speech and music signals in isolated and overlapped conditions. In real-life scenarios like movie audio sequences, speech and music are frequently encountered in overlapped situations. Standard spectrograms consist of a combined representation of harmonic and percussive striations. In the case of speech-music overlap, it might be challenging for automatic feature learning systems to extract class-specific patterns from such combined representations. Thus, this thesis proposes the use of the harmonic-percussive source separation method to generate features. The segregation of striation types is expected to improve the detection of speech and music signals. In addition, Multi-Task Learning (MTL) frameworks are also explored in this task that aid in training classification networks by simultaneously learning several related auxiliary tasks. Experimental results are reported on synthetically generated overlapped speech-music signals and natural recordings. Experiments show that harmonic and percussive decompositions of spectrograms perform better than the standard features. Moreover, classifiers with the MTL framework further improve the performance.

Movie trailer audio is predominantly composed of speech and music signals. The speech and music content vary according to movie genres. This thesis hypothesizes that information about speech and music signals in a movie trailer audio might relate to its genre label. Accordingly, the methods proposed in this thesis for speech and music detection are employed for movie genre classification. This is the fourth contribution of this thesis. An Attention-based Convolutional Neural Network classifier is proposed for the task. A large-size movie trailer dataset is used to benchmark the proposed approach against two baseline methods. It is observed that the proposed method performs better than the baselines. Moreover, the current proposal provides decent generalization performance.

This thesis has established the importance of audio modality in movie genre prediction. The speech-music identification methods proposed in this thesis have performed well on unseen movie audio signals. The decent success of this thesis is believed to spur further research to exploit other aspects of the audio modality in predicting movie genres.

Keywords: movie genre classification, speech music classification, spectral peak tracking, phase information, harmonic percussive source separation, multi-task learning

Contents

List of Figures

List of Tables

List of Acronyms

AAVW	Affective Audio-Visual Words
ALB	Attention Layer Block
ASE	Audio Segmentation Evaluation
ASPT	Amplitude Sequence of Peak Traces
CBoW-LSPT	CBoW feature computed from LSPT
CBoW-ASPT	CBoW feature computed from ASPT
CBoW	Component Bag-of-Words
CCA	Canonical Correlation Analysis
CLB	Convolutional Layer Block
CM-MCMN	Clean Music vs. MCMN
CNN	Convolutional Neural Network
CS-MCSN	Clean Speech vs. MCSN
DNGD	Double-differenced NGD
DFT	Discrete Fourier Transform
EF	Early-fusion
EM	Expectation-Maximization
FLB	Fully-connected Layer Block
GMM	Gaussian Mixture Model
GPF	GMM-Posterior Features
HMM	Hidden Markov Model
HNGD	Hilbert envelope of the Numerator of Group Delay
HNGDCC	MFCC computed from HNGD spectrum
HPF	Harmonic Percussive Features
HPS	Harmonic and Percussive Spectrograms

HPSS	Harmonic-Percussive Source Separation
kNN	k-Nearest Neighbors
IF	Intermediate-fusion
IFCC	Instantaneous Frequency Cosine Coefficient
IFQ	Instantaneous FreQuency
LF	Late-fusion
LHPS	Log-scaled Harmonic and Percussive Spectrograms
LMHS	Log-scaled Mel Harmonic Spectrogram
LMPS	Log-scaled Mel Percussive Spectrogram
LMS	Log-scaled Mel Spectrogram
LS	Log-scaled Spectrogram
LSF	Line Spectral Frequencies
LSP	Linear Spectral Pairs
LSPT	Location Sequence of Peak Traces
LTDM	Latent Topic Driven Model
MANOVA	Multivariate Analysis Of VAriance
MAP	Maximum A-Posteriori
MCSN	Mixed Class with Speech Noise
MCMN	Mixed Class with Music Noise
MFCC	Mel-Frequency Cepstral Coefficients
MGD	Modified Group Delay function
MGDCC	Modified Group Delay Cepstral Coefficient
MHS	Mel Harmonic Spectrogram
MHPS	Mel Harmonic and Percussive Spectrograms
MIREX	Music Information Retrieval Evaluation eXchange
MPEG	Moving Picture Experts Group
*mp*3	MPEG-2 Audio Layer III
MPS	Mel Percussive Spectrogram
ms	Miliseconds
MS	Mel Spectrogram

MSD	Mean and Standard Deviation
MSD-LSPT	MSD feature computed from LSPT
MSD-ASPT	MSD feature computed from ASPT
MSS	Multi-Speaker Speech
MTGC	Movie Trailer Genre Classification
MTL	Multi-Task Learning
MUSNOMIX	Music Speech Noise and Mixed
NAPS	Normalized Autocorrelation Peak Strength
NGD	Numerator of Group Delay
PAM	Peak Amplitude Matrix
PBF	Phase-Based Features
PLM	Peak Location Matrix
PLP	Perceptual Linear Prediction
RBF	Radial Basis Function
S	Gray-scale spectrogram
SF	Statistical-measure Features
SIM	Single Instrument Monophonic music
SMC	Speech vs. Music Classification
SMP	Speech vs. Music Prediction probabilities
SMR	Speech-to-Music Ratio
SNR	Speech-to-Noise Ratio
SPT	Spectral Peak Tracking
STL	Single-Task Learning
SVM	Support Vector Machine
TCN	Temporal Convolution Network
VGG	Visual Geometry Group
VLAD	Vector of Locally Aggregated Descriptors
VSD	Violent Scene Detection
ZCR	Zero Crossing Rate
ZFFS	Zero-Frequency Filtered Signal

List of Symbols

$\mathbb{R}$	Set of real numbers
$x[n]$	An audio signal with where $n = 0, \ldots (N_s - 1)$
N_s	Number of samples in $x[n]$
L	Total number of overlapping short-term frames extracted from $x[n]$
x_l	A short-term frame of length N
t_w	Duration of x_l in miliseconds
t_s	Shift of x_l in miliseconds
F_l	Feature vector corresponding to x_l
$F_\mu^{(l)}$	Mean of $2W$ feature vector around F_l
$F_\sigma^{(l)}$	Standard-deviation of $2W$ feature vector around F_l
$\mathcal{R}^D$	The set of D-dimensional real numbers
N_f	Half of the number of DFT bins
$\mathbf{X}_l$	DFT of x_l, $l = 0, \ldots (L-1)$
e	The exponential symbol
π	Pi
$\mathbf{H}_l$	Frequency locations of all spectral peaks in X_l
$\tilde{\mathbf{H}}_l$	Frequency locations of highest p spectral peaks in X_l
$\mathbf{fH}_l$	$\tilde{\mathbf{H}}_l$ sorted in descending order
$\mathcal{L}$	$p \times L$ Peak Location Matrix
$\mathcal{A}$	$p \times L$ Peak Amplitude Matrix
$\mu_r^{\mathcal{L}}$	Mean of r^{th} LSPT, $r = 0 \ldots (p-1)$
$\sigma_r^{\mathcal{L}}$	Standard deviation of r^{th} LSPT, $r = 0, \ldots (p-1)$
$\mu_r^{\mathcal{A}}$	Mean of r^{th} ASPT, $r = 0 \ldots (p-1)$
$\sigma_r^{\mathcal{A}}$	Standard deviation of r^{th} ASPT, $r = 0, \ldots (p-1)$

$\mathcal{G}$	A GMM
$\mathcal{C}_j$	The j^{th} mixture component of a GMM, $j = 0, \ldots (K-1)$
μ_j	Mean of $\mathcal{C}_j$
ν_j	Variance of $\mathcal{C}_j$
$\mathcal{P}\left(\mathcal{C}_j^r \mid u\right)$	The posterior probability of the jth component $\mathcal{C}_j^r$ of $\mathcal{G}^r$ with respect to u
$P\left(u \mid \mathcal{C}_j^r\right)$	Gaussian likelihood function
${}_s\mathcal{L}^t$	The PLM matrix constructed from the t^{th} interval, $t = 0, \ldots (T_s - 1)$ of speech training data
${}_m\mathcal{L}^{\tilde{t}}$	The PLM matrix constructed from the $\tilde{t}^{th}$ interval, $\tilde{t} = 0, \ldots (T_m - 1)$ of music training data
${}_{ls}\mathbf{S}^r$	Set of frequency locations of the rth peak traces of speech training data
${}_{lm}\mathbf{S}^r$	Set of frequency locations of the rth peak traces of music training data
${}_{ls}\mathcal{H}^r$	Averaged posterior probability vector for speech data
${}_{lm}\mathcal{H}^r$	Averaged posterior probability vector for music data
γ_{rbf}	Bandwidth of Radial Basis Function kernel of Support Vector Machines
sr	Sampling Rate
$\tau[k]$	Group delay function
n_{mel}	Number of Mel filters
$\tau_{MGD}[k]$	Modified Group Delay function
C	Number of narrow band components used for computing IFCC
$\mathcal{F}_{\mathcal{D}}^{-1}$	Inverse DFT operator
Λ	Wilks' Lambda
$\mathbf{X}_H$	Harmonic decomposition of DFT spectrogram $\mathbf{X}$
l_{harm}	Median filter size for harmonic decomposition
$\mathbf{X}_P$	Percussive decomposition of DFT spectrogram $\mathbf{X}$
l_{perc}	Median filter size for percussive decomposition
$\mathbf{M}_H$	Soft-mask for harmonic decomposition
$\mathbf{M}_P$	Soft-mask for percussive decomposition
n_t	Numbers of frames in a spectrogram patch used as a classification unit
$s_R^{(i)}$	Skewness computed from the i^{th} row of a spectrogram; $i = 0, \ldots 20$
$\mathbf{r}_{skew}$	Row skewness vector
$s_C^{(j)}$	Skewness computed from the j^{th} column of a spectrogram; $j = 0, \ldots 67$

$\mathbf{c}_{skew}$	Column skewness vector
AT_S	Speech vs. non-speech auxiliary task
AT_M	Music vs. non-music auxiliary task
AT_R	SMR estimation auxiliary task
ϕ	2-dimensional target for the AT_R auxiliary task
ϕ_M	Music scaling factor
ϕ_S	Speech scaling factor
$\mathcal{L}_s$	Loss function of AT_S auxiliary task
y_s	Ground truth for AT_S auxiliary task
$\hat{y}_s$	Predicted output of AT_S auxiliary task
$\mathcal{L}_m$	Loss function of AT_M auxiliary task
y_m	Ground truth for AT_M auxiliary task
$\hat{y}_m$	Predicted output of AT_M auxiliary task
$\mathcal{L}_{smr}$	Loss function of AT_R auxiliary task
y_{smr}	Ground truth for AT_R auxiliary task
$\hat{y}_{smr}$	Predicted output of AT_R auxiliary task
N_B	Number of samples in a training batch
$\mathcal{L}_c$	Loss function of music/speech/speech+music classification main task
y_c	One-hot encoded ground truth of the main task
$\hat{y}_c$	Predicted output of the main task
$\mathcal{L}_{Total}$	Total loss of the MTL-based architecture
CM	Confusion matrix
p_{harm}	Prediction of model trained on the harmonic feature
p_{perc}	Prediction of model trained on the percussive feature
α_{LF}	Parameter used to combine the predictions from two classifier models in LF strategy
$U_{\mathcal{A}}$	12-dimensional statistical features computed from peak amplitude sequences
$U_{\mathcal{L}}$	12-dimensional statistical features computed from peak location sequences
$\mathbf{C}$	A $n_t \times n_c$ output of the CLB of ACNN classifier
n_c	Number of filters in the last convolution layer of CLB
$\hat{W}$	Perceptron used to train the attention layer

$\hat{h}$	Hidden representation learned in the attention layer
$\hat{u}$	Perceptron used to learn context weights in the attention layer
$\hat{\alpha}$	Context-weights produced by the attention layer
$\hat{\mathbf{c}}$	Context vector obtained after applying time-axis attention
$\tilde{\mathbf{c}}$	Context vector obtained after applying channel-axis attention
$\mathbf{c}$	Concatenated time and channel attention context vectors
$V_{\mathcal{A}}$	Averaged CBoW-ASPT features computed from movie trailer segments
$V_{\mathcal{L}}$	Averaged CBoW-LSPT features computed from movie trailer segments
$U_{\mathcal{A}}$	Statistical features computed from spectral peak amplitude tracks of movie trailer segments
$U_{\mathcal{L}}$	Statistical features computed from spectral peak location tracks of movie trailer segments

1

Introduction

Contents

Objective

Movie viewership has increased many folds over the past decades. The rise in internet usage and the advent of Over-The-Top (OTT) media platforms have further proliferated the per-capita consumption of movies and other audio-visual documents. Manual analysis of such large volumes of movie content is a highly challenging task. Thus, the development of automatic methods to analyze the ever-increasing movie content has become more necessary than ever. One of the essential tasks among various applications is the automatic determination of movie genres. Movie genre information might be helpful in underage censorship, choice-based query retrieval, and targetted publicity. This thesis aims to develop systems for the automatic determination of movie genres. More specifically, the audio modality of short trailers of movies is used to predict their associated genre labels. This thesis is arguably the first work of its kind where a detailed study of the audio modality is performed for movie genre classification. The movie audio mainly comprises speech and music signals in both isolated and overlapping scenarios. This thesis pursues the hypothesis that the presence of speech and music signals in movies might vary according to their associated genres. Hence, detecting speech and music in movie audio might help in genre identification. Therefore, a detailed analysis of these audio signals is performed in this thesis. In this regard, features and classification models for efficient speech and music signal classification are also proposed. The proposed approaches improve the speech-music detection performance compared to that of the state-of-art methods from the literature. Subsequently, the proposed speech and music detection methods are further employed in the movie genre classification task. The performances of the proposed methods in movie genre classification are found to be satisfactory and compare well with state-of-the-art. The obtained results validate the hypothesis of this thesis.

1.1 Automatic processing of movies: An overview

Movies have become an integral part of public entertainment. Since making the world's first motion picture (Roundhay Garden Scene, 1888), there has been tremendous growth in the movie industry. The number of movie releases per year, production quality, use of Computer Generated Imagery (CGI), Visual Effects (VFX), viewership, and revenue generated by the movie industry has consistently risen. The year-wise increase in the domestic revenue of the Hollywood industry from 1977 to 2022 is illustrated in Fig. 1.1(a). The annual revenue growth of the Indian movie industry

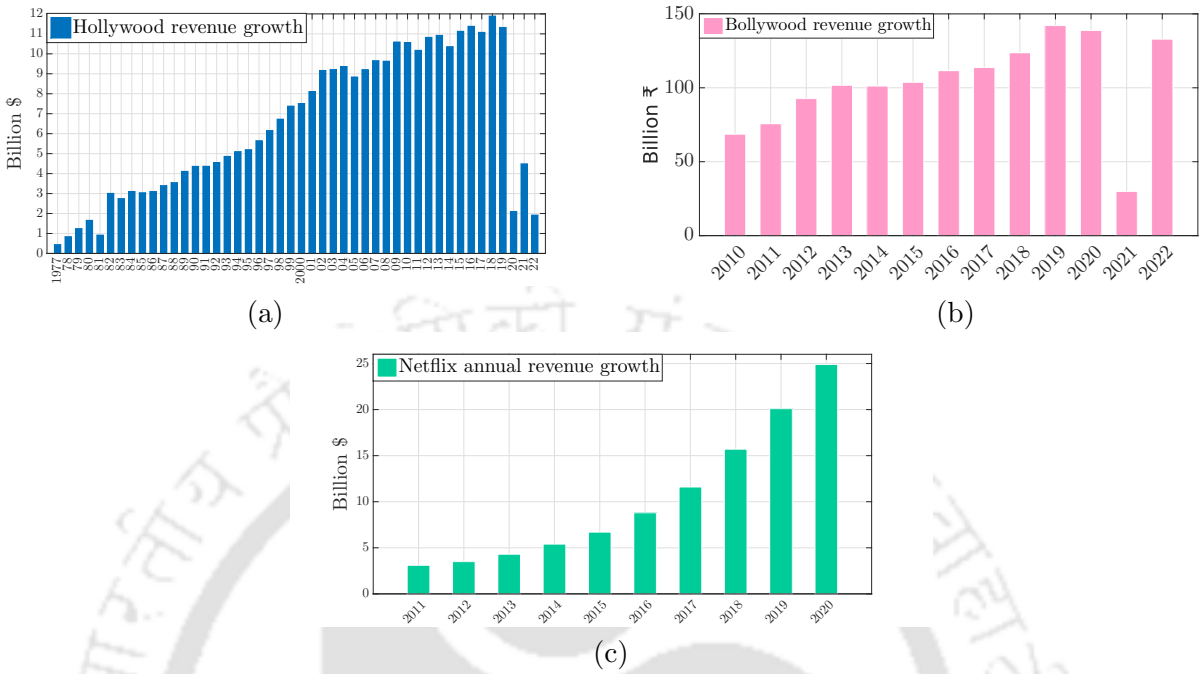


Figure 1.1: Illustrating the (a) Hollywood annual revenue growth from 1977-2022 (Courtesy: `www.boxofficemojo.com`), (b) Bollywood annual revenue growth from 2010-2022 (Courtesy: `statista.in`), and (c) Annual revenue growth of Netflix from 2011-2020 (Courtesy: `selectra.in`)

(Bollywood) is illustrated in Fig. 1.1(b). Barring the pandemic years of 2020-21, the revenue of both Hollywood and Bollywood movie industries has been recording an overall growth. Similar trends might be observed in the other movie industries across the world. Such trends indicate that the consumption of movies is set to increase further in the future.

A relatively recent development is the rise of Over-the-Top (OTT) media. The OTT platforms like Amazon Prime, Netflix, and others are enjoying ever-increasing viewership and popularity [3, 4]. Fig. 1.1(c) shows the consistent growth in the worldwide annual revenue of Netflix from 2011 to 2020. It has been predicted that broadband speeds, the number of internet users, and per-capita mobile devices will rise significantly in the upcoming years [5]. Such trends would further augment the proliferation of streaming content consumption. Efficient archival, search, and retrieval of enormous video databases can no longer be performed manually. Strategies for automatic processing of such content have become a need of the hour. The upcoming subsection describes the various research problems being explored in the context of movies and the goal of this thesis.

1.1.1 Computational research problems in movies

Movies have witnessed significant research attention in the past. Researchers have attempted to develop efficient solutions for various practical problems related to movies. Some of the broad categories of such tasks are listed below.

(i) ***Movie Analytics*** deals with interpreting the movie content and studying its intrinsic structure for purposes related to film education, entertainment, or research [6, 7].

(ii) ***Movie Recommendation*** helps viewers easily find movies of their preferences from huge databases [8, 9] (for example *MovieLens* [1]).

(iii) ***Movie Success Forecasting*** involves the development of models that predict the amount of box-office business a new movie may generate. Such predictions can help in making informed decisions about developing related products and logistics [10, 11].

(iv) ***Movie Abstraction*** is the task of converting the content of a movie to a condensed form that eases browsing or searching by viewers. Abstraction may be performed by generating a sequence of essential video frames (termed summarization) [12] or by creating a short representative video clip (termed skimming) [13].

(v) ***Movie Affective Content Analysis*** deals with generating semantically meaningful movie metadata by automatically identifying scenes that might stimulate a specific emotional reaction from viewers [14, 15].

(vi) ***Movie Speaker Identification*** deals with identifying actors from their highly emotive movie dialogues [16, 17]. This task is slightly different from traditional speaker recognition due to the challenging speaking modes in movies.

(vii) ***Acoustic Event Diarization*** in movie audio is an integral part of many multimedia analysis systems. This task involves the identification of non-silent sounds in audio, segmenting different acoustic events, and labeling them [18].

(viii) ***Movie genre classification*** tackles the problem of mapping diverse signal-level information of movies to human interpretation-based semantic concept like genre [19, 20].

[1] https://movielens.org/

Movies are an inseparable part of the cultural consciousness of people. With the development of modern technology, the creation and delivery of movies to viewers are being enhanced tremendously. The various contemporary movie-based research problems (listed above) are evidence of the necessity of improving audiences' searching and viewing experience. Researchers have explored different modalities associated with movies, viz., audio, video, posters, and various meta-data like plot summaries for various tasks. Lu et al. [21] mentioned that the knowledge of one modality might be used to understand the contents of another. Also, the relationships between modalities may be utilized for effective retrieval of one modality using a query made in another [21]. In this context, the audio modality of movies is relatively less studied in the literature. Audio is a crucial component of movies that must be exploited for various tasks. Hence, this thesis is mainly concerned with the audio modality to fill the void. The upcoming section describes the importance of the audio component of movies.

1.2 Movie audio characteristics

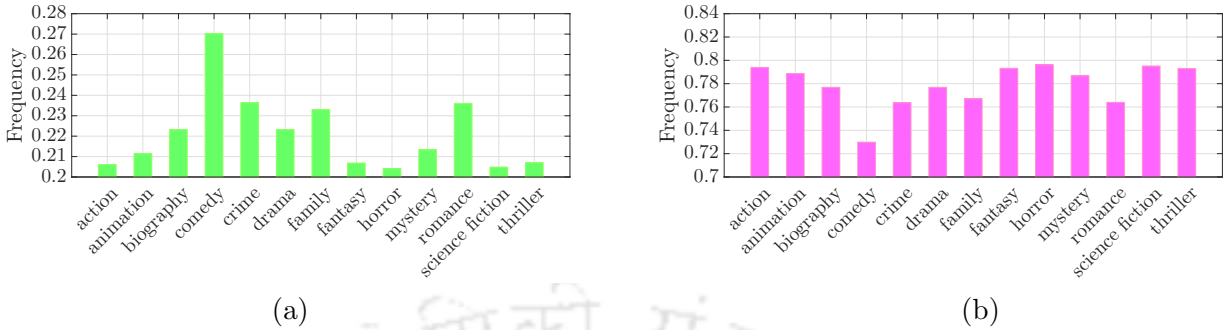
The audio component of movies is rich in information about the movie content. Audio-based processing of movies might aid in understanding aspects that the visual modality may not otherwise reveal. For example, Bougiatiotis et al. [22] mentioned that a movie's "style" is determined in large part by its audio content. Movie editors frequently employ specific sounds and music to highlight emotional situations and dramatic effects [23]. The use of different mood-specific sounds also varies according to culture. Director's intent and style can often be elucidated by the music used in the movie [24]. Stylistic editing and mixing of the movie audio are performed in the post-production stage of a movie. Such sound editing approaches are often motivated by the need to evoke specific emotions in the audience [25]. Speech content in movies frequently differs from everyday conversational speech. It contains normal, whispered, and shouted speech along with singing and electronically modified speech [25]. Irie et al. [15] noted that some features in movie audio have a strong relation to affective scenes. The affective information is carried by the music component [26] and connotative aspects like shooting and editing conventions [27]. Thus, the audio component deserves to be better analyzed to comprehend its effect on the different aspects of movies. The upcoming subsections discuss the structural components of the movie audio.

1.2.1 Composition of movie audio

The movies were just motion pictures without any sound in the initial days. The first feature-length movie ever made in the world was *The Story of the Kelly Gang* (1906). It was released in the Athenaeum Theatre in Melbourne. It did not have any audio components. Voices and sound effects were played live during the film's screening. The first non-silent movie ever made was *The Jazz Singer* which premiered on 6^{th} October 1927 in the United States of America. Even though it had only a few sound sequences, movies of this type eventually came to be known as *Talking Pictures* or *Talkies*. This movie effectively marked the end of silent-era films. The talkies became extremely popular worldwide by the early 1930s. The inclusion of sound in movies dramatically increased the cost and complexity involved in movie making. Initially, many moviemakers were worried about the aesthetic quality of movies being degraded because of the inclusion of dialogues. Some countries like Japan were slow in adopting the new form of sound-embedded movies. In comparison, other countries like the United States of America and India lapped up the opportunity from the very beginning. Since then, there has been a tremendous overhaul in the movie-making process.

Audio has now become an integral part of movies. The development of surround sound technology established the indispensability of audio. This technology allows moviemakers to present sound in a three-dimensional model to generate effects that enhance the audience's viewing experience by letting them relive a particular movie scene. Such is the power of the audio component. The movie audio is now an extremely complex signal composed of varied sound sources. Bougiatiotis et al. [22] noted that the movie soundscape consists of music, speech, sound effects, and environmental acoustic events that combine to form a vividly rich signal. Understanding the type and quality of sounds present in movie audio can help in automatically processing the movie content. Therefore, the speech and music signals need to be analyzed in the context of movie audio.

Speech is one of the major components of movie audio. As spoken dialogues of actors or narrators, speech is used to carry forward the movie storyline. Movie audio has a diversity of speakers and speaker demographics [29]. Moreover, the speech of the same actor varies across different scenes in the same movie to enact the portrayal of diverse plots. Actors use various vocal modes other than normal conversational speech, like whispering and shouting. The scene's genre influences the prosodic quality of speech like pitch, speaking rate, loudness, and others. For example, the speech in a horror scene might be laced with fear and disgust. In comparison, the speech in a comedy scene might be



(a) (b)

Figure 1.2: Illustrating the genre-wise distribution of (a) Speech, and (b) Music signals across the trailers of *Moviescope* dataset [28] predicted using the speech vs. music classifier developed in [1].

full of joy and excitement. Detection of speech regions or Voice Activity Detection (VAD) in movies is an ongoing research problem. The VAD is essential for various applications like subtitle generation and audio diarization, to name a few. The presence of a significant amount of non-speech segments and speech with background sounds make the VAD task quite challenging [30]. Fig. 1.2(a) shows the genre-wise distribution of predicted speech in the *Moviescope* dataset consisting of trailers of approximately 5000 movies. The speech predictions are generated using a recently proposed speech vs. music classification system [1]. Fig. 1.2(a) indicates that biography, comedy, crime, drama, family, and romance genres correspond to significantly more speech segments compared to other genres. Thus, it can be inferred that the distribution of speech across genres varies. Such information may be utilized while predicting movie genres.

Music is arguably the most significant component of movie audio. Different emotional responses can be elicited in viewers by the presence of music in movie audio [31]. Music may be present as isolated instrumental pieces or can be accompanied by vocals. Some segments in movies may consist of loud music as the sole audio component. Other segments may contain background musical scores with dialogues to highlight the mood of specific scenes. Music genre classification [32] and singing voice separation [33] are some of the contemporary research problems related to music information retrieval. The music used in movies is heavily influenced by the prominent artistic traditions of the moviemakers. For example, Indian movies contain vocal music as part of the movie narrative. The music compositions used in the different regional movies in India are influenced by their respective cultures. Music also plays a part in highlighting the genre of a movie. For example, action and horror genres correspond to fast-paced music, while romance and drama genres are associated with slower sentimental musical tracks [24]. Fig. 1.2(b) shows the distribution of predicted music segments

across different genres of the *Moviescope* dataset. As mentioned earlier, the music predictions are generated using the classifier developed in [1]. As can be observed in Fig. 1.2(b), the biography, comedy, crime, family, and romance genres consist of the least amounts of music among other genres. Such a distribution might be because of the prevalence of dialogues in these genres. Nevertheless, all genres contain significant amounts of music (Fig. 1.2(b)). Thus, the detection of music signals might be useful in determining the genres of movies.

Apart from speech and music, movie audio consists of various environmental sounds, sound effects, and noises. These sounds give a feel of the acoustic environment of a particular movie scene. For example, a courtroom scene would consist of the ambient sounds generally found in a courthouse, like sounds of the judge's gravel, the rustling of papers, and the scuttling of chairs. Similarly, a scene in a public park might consist of sounds like birds chirping, the rustling of dried leaves, and noises made by people. Such soundscapes are created to provide the audiences with a feel of the happenings in the movie. Sound effects are not necessarily a part of the movie narrative, but they play a vital role in dramatizing specific scenes. Background noise is often present in movie audio that can drown out various sound events like dog bark or gunshots [34]. Thus, detecting such sounds in movie audio is a challenging job. Hoiem et al. [35] noted that sound events like a dog bark, gunshots, car horn, doorbell, and others are scarcely distributed in movie audio. Therefore, such sounds in isolation might not be very informative about the movie genre. However, they may help in predicting movie genres in combination with the music and speech components.

1.3 Motivation for movie genre classification

Movies have been a popular research topic for the community. Various prominent directions of exploration that fellow researchers are currently pursuing are listed in section 1.1.1. This thesis aims to develop methods for efficient genre classification of movies that can aid in building better searching, retrieval, and archival applications. The IMDb website [2] defines *genre* as "a category of artistic composition characterized by similarities in form, style, or subject matter for a piece of content". Genre is one of the favored metadata used by audiences for choosing movies for viewing. Genre information of movies also helps develop efficient recommender systems [9]. This information may help viewers choose movies of non-native languages for which reviews are unavailable. Currently, genres are assigned by either the creative departments of movie production houses or film critics, media

[2] www.imdb.com

and audiences. However, such approaches of manually associating movies with relevant genre labels is a challenging data annotation task. It requires the involvement of many people for the reliability of annotation. Also, it is affected by the personal subjectivity of annotators. With the increase in the number of movies produced, manual assignment of genre labels may not be feasible anymore. Thus, the availability of efficient automatic genre classification systems for movies might benefit viewers searching for content of interest and the movie makers or distributors targetting specific audiences for publicity. Hence, this thesis is focused on developing methods that can effectively map the signal-level information of movies to their respective genres. Recently, automatic genre classification of movies from their short (approximately 1 to 3 minutes) trailers has attracted many researchers. Movie trailers are designed in a particular manner so that different emotional responses may be evoked in the viewers [36]. Trailers usually contain rich and varied content that represents the theme of the actual movie. Therefore, this thesis proposes systems for efficient genre classification of movies from their short trailers.

The importance of the audio modality of movies is described in section 1.2. Audio plays a critical part in conveying the story presented in a movie. Audio also appears to contain a significant amount of genre-specific information. Moreover, the affective characteristic is said to be better characterized by audio than video [37]. Such observations indicate that the audio modality needs to be studied in more detail for better movie content analysis. In addition, the real-time performance of applications like movie recommender systems that depend on genre information is a significant requirement [8]. Bougiatiotis et al. [22] noted that the visual domain-based applications are computationally more demanding than audio. Therefore, the audio-based processing of movies might be beneficial from the content analysis aspect and the real-time performance point of view.

This thesis aims to develop various signal processing algorithms to process the movie trailer audio and assign probable genres. As discussed previously, speech and music are significant components of movie audio. Thus, it is imperative to identify these signals efficiently. The information about speech and music signals detected in movie trailer audio may be utilized to determine the movie's genre. Therefore, the first goal of this thesis is to develop effective Speech vs. Music Classification (SMC) systems for movie audio. The acoustic characteristics of speech and music are distinctly different. Many researchers have attempted this task previously in the literature with exceptional success (described in chapter 2). Nevertheless, we have identified some approaches to target this

problem that was not explored previously. This thesis aims to propose novel features for the SMC task that have not been previously explored in the literature (discussed in chapter 3 and 4). Furthermore, speech and music are frequently found as overlapping mixtures in movies. The performance of standard SMC methods can be expected to degrade in the presence of speech+music signals. Thus, this thesis also proposes better features and classifier designs for the efficient detection of speech+music signals. Subsequently, the various speech-music detection systems developed in this thesis are applied to the task of genre prediction of movie trailers. This thesis hypothesizes that the use of speech and music information may be beneficial for the task of movie genre classification. The organization of this thesis is described next.

1.4 Thesis organization

The organization of this thesis is described below.

- Chapter 2 describes the existing literature on movie genre classification. The present thesis proposes to employ speech and music information in predicting movie genres. Thus, the available work on SMC and speech+music detection are also reviewed in chapter 2.

- Chapter 3 describes the *Movie-MUSNOMIX* dataset contributed through this thesis. The *Movie-MUSNOMIX* dataset consists of audio signals from four Indian movies annotated with seven audio categories, including speech and music. The chapter also proposes novel spectral peak tracking-based magnitude spectrum features for the SMC task. The speech spectrograms contain curvy striations, whereas music has linear striations. Motivated by such an observation, a novel spectral peak tracking algorithm is developed to capture such patterns from spectrograms. Subsequently, two novel features are designed that are observed to perform very well in the SMC task.

- Chapter 4 proposes to use the relatively less investigated phase information for the SMC task. Three existing phase-based features are explored in this context. These features were originally proposed for other speech-related tasks. The phase-based features carry information that is complementary to the magnitude-based ones. Hence, combining these two types of features improves SMC performance.

- Chapter 5 investigates the task of detecting speech+music signals. The previously unexplored

Harmonic-Percussive Source Separation (HPSS) based features are employed for the task. Additionally, the successful Multi-Task Learning (MTL) framework has been employed in designing better classifiers for the task. The HPSS-based features and the MTL-based classifiers are found to improve the classification performance when compared with state-of-the-art methods from the literature.

- Chapter 6 performs the Movie Trailer Genre Classification (MTGC) task using the methods developed in the previous chapters. A novel Attention-based Convolutional Neural Network (ACNN) classifier is designed for the MTGC task. The features developed in each earlier chapter are individually applied to the MTGC task with the ACNN classifier. Finally, all the systems obtained from each chapter are combined to obtain the best MTGC system. The results obtained validate the hypothesis that speech and music signals carry genre-specific information about the movies. Therefore, efficient genre prediction of movies can be achieved by using the information about speech and music in the movie audio.
- Chapter 7 summarises the thesis and discusses the possible directions of future extensions of the work presented in this thesis.

2

Literature Review

Contents

Objective

One of the crucial tasks of automatic movie processing is genre classification. Genre is a human interpretation-level semantic used to group movies according to their similarity in style, content, and presentation. Previous works have primarily focussed on the visual modality of the movies to predict the genre. Audio has been predominantly used as supplementary information to aid the visual feature based approaches. There is limited analysis of the movie audio modality for genre classification in the existing literature. This thesis attempts to develop an efficient audio-based movie genre classification system. In this regard, this chapter reviews the available literature on movie genre classification approaches. Furthermore, the movie audio contains speech and music signals as the most significant components. Any audio-based genre classification method is expected to identify movie audio components efficiently. Therefore, a detailed study of the methods for detecting speech and music in isolation and overlapping scenarios is necessary. This chapter also provides a study of the available speech and music detection approaches. Finally, the lacuna in the available literature is highlighted to motivate the present thesis.

2.1 Movie analysis: A review

Movie-makers generally release short trailers before the actual premiere of the entire movie. These trailers have rich and varied content that represents the actual movie. The task of publicizing a movie before its official release is served by the trailers. The trailers provide glimpses of the movie by alluding to the significant events while keeping the plot in suspense. Yadav et al. [36] noted that the design of the movie trailers is such that they elicit a range of emotional responses in the viewers. Trailers are intentionally created to spike the viewers' interest in watching an upcoming movie aligned to their genre of liking. Most existing works do not use full-length movies for their genre classification. Instead, their trailer videos are predominantly used for the task. The use of movie trailers in this task may be reasoned as follows. First, the movie-makers create trailers as a concise abstraction of the entire movie. Thus, they must consist of the most relevant information regarding the movie's genres. Second, the computational requirement of processing a short trailer video is significantly lower than the requirement of processing a whole movie. Accordingly, this thesis also uses movie trailers for performing genre prediction. This chapter reviews the relevant literature on movie genre classification.

2.1.1 Affective understanding in movies

Affective understanding in movies is an important related area of research that aims to predict the emotional response evoked in viewers upon watching a movie. Genres can be viewed as the emotional response expected from a movie. On the other hand, affective labels determine the actual emotion felt by the viewers. Even though both the labeling schemes have similarities, they may differ in their applications. Nevertheless, a review of works attempting to predict the affective labels of movies might be helpful.

Wang et al. [37] described that a semantic gap exists between low-level features and high-level human perceptions like affective characteristics. Mid-level features may be explored to bridge this gap [37]. Such mid-level features can be manually defined [27] or automatically learned from data [15]. Wang et al. [37] noted that a combination of manual and automatically learned mid-level representations would be a promising approach for affective classification.

Wang et al. [38] attempted to bridge the semantic gap between low-level features and high-level emotions by proposing the use of psychology and cinematographic information for the affective understanding of movies. The authors employed audio-visual features in the task. First, the audio was segmented into scenes and detected the audio type of 2s clips within scenes. The authors considered four audio types, viz., MUSIC, SPEECH, ENVIRONMENTAL SOUNDS, and SILENCE. Energy thresholding-based silence detection was employed. Music, speech, and environmental sounds were detected with an SVM classifier. Subsequently, the ratio of the relative duration of each audio type to the scene length was computed and termed as Audio Type Proportion. Various features computed from the detected audio types were then used to determine their affective content. A Scene Affect Vector was computed that quantified the proportion of various affect categories in a scene. The authors considered six affect categories, viz., ANGER, SADNESS, FEAR, JOY, SURPRISE, and NEUTRAL. Similar processing was performed for the visual data. It was observed that audio cues were more informative than visual cues for determining affective content.

Soleymani et al. [39] ranked movie scenes based on the natural physiological response of the participating viewers. Irie et al. [15] performed affective scene classification by determining the expected emotional response of viewers to given movie scenes. An Affective Audio-Visual Words (AAVW) was proposed as a feature. The classification was performed by using a Latent Topic Driven Model (LTDM), previously proposed in [40] for affective scene classification. Authors described emotion as

a Markovian dynamic system that triggers as the movie content progresses. The proposed LTDM had two sub-models, viz., a topic model and an emotion model. The emotion model estimated the sequence of emotions from the representative AAVW components extracted by the topic model using latent Dirichlet allocation. The audio features used in computing AAVW were pitch, short-term energy, and Mel-Frequency Cepstral Coefficients (MFCC).

Benini et al. [41] tried to map the low-level audio-visual properties of movies to the high-level emotional responses evoked in the viewers. The connotative properties of movies were proposed to perform affective movie recommendations. Connotative properties refer to the shooting and editing conventions followed during movie development. Their proposed connotative space is defined by natural, temporal, and energetic dimensions.

Canini et al. [27] used Connotative information as mid-level features to predict the emotional response of users to movies. The authors employed audio-visual information in the task. Authors defined connotative properties as "the set of shooting and editing conventions that help transmit meaning to the audience". The connotative space is described in three dimensions, viz., energetic/minimal, cold/warm, and slow/dynamic. Authors argue that connotative properties were more agreed upon than emotional responses by people. The MFCC was used along with other standard audio features, visual features, and grammar information.

Xu et al. [42] proposed a hierarchical method to determine movie emotion by first detecting emotion intensity or arousal (a continuous scale from low to high). The C-Means clustering was performed over arousal-based features, followed by detection of emotion type using valence-based features. Valence is defined on a continuous scale from negative to positive emotions. The authors used Audio-visual features. Short-term energy and MFCC were used as audio-based arousal features. Pitch was used as the audio-based valence feature. The temporal characteristics of movie emotions were captured using a Conditional Random Field-based system. Four movie genres were considered, viz., ACTION, HORROR, DRAMA, and COMEDY.

Both categorical classifiers and continuous regression-based models have been explored in literature for video affective content analysis [37]. The features learned in Deep-Learning (DL) framework were found to be more successful than the ones engineered based on cinematography domain knowledge. However, years of understanding cinematography and physiological knowledge were helpful with data-driven approaches. Tarvainen et al. [43] performed acoustic scene classification of movies into categories

like INTERIOR/EXTERIOR, DAYTIME/NIGHTTIME, and others. The authors observed that detecting the amount of speech and music in movie audio was very useful for scene detection. Hence, music emotion was used as an additional feature with image features for the task.

2.1.2 Violent scene detection

Among the initial works related to movie genre classification, Violent Scene Detection (VSD) has been a critical application that researchers have explored. Violence is defined as the intentional infliction of physical harm or threatening thereof from people against people [44]. Detection and censorship of violent scenes are necessary to protect sensitive social groups like children [45]. This subsection reviews VSD approaches available in the literature.

Giannakopoulos et al. [45] performed VSD using the audio modality. The audio stream of movies was classified into six sound types. The identified sounds were further grouped into either VIOLENT or NON-VIOLENT categories. Music, speech, and non-violent environmental sounds were considered NON-VIOLENT sounds. Whereas shots, fights, and screams were categorized as VIOLENT sounds. The authors used 12 standard audio features apart from MFCC for the task. The classification was performed using Bayesian Networks.

Giannakopoulos et al. [46] performed VSD by fusing audio and visual features. Seven types of audio signals were identified and categorized into VIOLENT and NON-VIOLENT groups. Music, speech, low-energy environmental sounds (like silence, background noise, and others), and abrupt sound impulses (like a door closing, thunder, and others) were considered NON-VIOLENT audio types. Whereas shots, fights, and screams were considered VIOLENT audio types. The authors used a combination of k-Nearest Neighbor (kNN) and Bayesian Networks to generate a 7-class probability for each audio frame using 12 standard audio features. Similarly, authors generated 2-diemensional probabilities for low and high activity from 1s mid-term visual segments. An early fusion of audio-based and video-based probability vectors was performed with a kNN classifier to determine VIOLENT scenes.

Souza et al. [47] performed VSD using local spatio-temporal features with a bag of visual words and a linear-kernel SVM classifier. Whereas Chen et al. [48] performed VSD in a two-step process. The ACTION scene was detected first. Subsequently, the face, blood, and motion information were incorporated to categorize it as a HORROR scene. The SVM was used for classification. Acar et al. [49] showed that mid-level audio features in the form of Bag of Audio Words computed from MFCC features perform better than low-level audio and visual features in VSD.

A recurring task in the popular *MediaEVAL* Benchmarking campaigns from 2011-2015 [50–54] was VSD. The main purpose of the VSD task was to propose a common public evaluation framework for violence detection. The task entails automatic detection of VIOLENT scenes and the emotional impact of short video clips using multimedia features. Constantin et al. [55] provided a detailed review of the previous editions of the VSD task in *MediaEVAL* campaigns. The authors curated all datasets released for the *MediaEVAL* campaigns into a combined $VSD96$ corpus [55] of 96 hours. The submissions to *MediaEVAL* have explored different approaches to detect violent scenes. Dai et al. [56] explored multi-modal features in the task. Derbas et al. [57] utilized traditional visual and audio features. Features based on audio and different concepts (like blood, explosions, gun-shots, screams, etc.) were used by Penet et al. [58], while Schlüter et al. [59] employed visual, audio, and conceptual features. Embeddings generated from deep networks were also explored [60]. A combination of audio-visual features and audio-conceptual features was the best feature fusion. Such results emphasized the importance of audio modality in the task. It was also observed that early and late fusion of features were the best performers. The MFCC remained the most popular among different standard temporal and spectral audio features explored by the participating systems. A variety of classification systems were employed by the participants, like SVM [61], Bayesian Networks [58], shallow Neural Networks [59], Deep Neural Networks [56], Discriminant Analysis [62], kNN [63], unsupervised methods [64] and hybrid methods [57]. The SVM was found to be the best performing classifier in the VSD task. The movie genre classification literature is reviewed next.

2.1.3 Movie genre classification

One of the initial works in movie genre classification was done by Rasheed et al. [65]. The authors classified movie trailers into multiple hierarchical genres using audio-visual features. Peakiness in the audio energy plot was used as one of the features. At the first level, ACTION and NON-ACTION movies were classifier. Then, ACTION movies were classified into gunfire/explosions and others at the second level. NON-ACTION movies were classified into COMEDY, HORROR, and others. In another work of the authors, Rasheed et al. [66] performed movie trailer classification into COMEDY, ACTION, DRAMA, or HORROR genres using only low-level visual features and Mean-shift clustering. Whereas Jain et al. [67] used early-fusion of audio-visual features with a feed-forward neural network to classify movies into five genres, viz., ACTION, HORROR, COMEDY, MUSIC, and DRAMA. The pitch, spectrum-based, energy, and MFCC were used as audio features.

Austin et al. [24] proposed the use of musical scores to categorize movies into ROMANCE, DRAMA, HORROR, and ACTION genres. The authors used MFCC, Linear Prediction Coefficients (LPC), Zero-Crossing Rate (ZCR), and other standard spectral features to represent timbral information. Also, tempo, beat, and other rhythm features were used in the task. The combined feature vector used in the task had 222 dimensions. SVM classifiers were used to perform pair-wise and four-class classification of genres.

Zhou et al. [68] performed movie trailer genre classification with high-level scene detection on keyframes extracted from the trailers. A video shot is defined as a set of consecutive frames having similar image feature distribution. The video shots were categorized using the scene detection features in an unsupervised manner. The category information of all shots in a trailer was used as a bag of visual words feature. Authors mapped the trailers into one of four genres, viz., ACTION, COMEDY, DRAMA, or HORROR.

Wang et al. [23] performed HORROR scene detection using Multiple Instance Learning (MIL). Visual, aural, and high-level image features representing emotions for each scene were utilized. The authors considered a video scene to consist of consecutive video shots. Each video scene was considered as a bag. The features extracted from each component shot were used to represent the video scene as a Bag of Shots. The frame-wise MFCC, power, spectral centroid, and ZCR were used as the audio features. In a previous work of the authors [69], they proposed the use of image-based emotional features along with visual and aural features for HORROR scene detection using an SVM classifier.

Huang et al. [70] performed movie genre classification using an ensemble of one vs. one Support Vector Machine classifiers with Radial Basis Function kernels for each genre-pair. The authors used a 277-dimensional feature created as a combination of audio and visual features. Spectrum, compactness, root-mean-squared energy, ZCR, LPC, MFCC, and rhythm were used as the audio-based features. Feature selection for each genre pair was performed using Self-Adaptive Harmony Search [71]. The task considered ACTION, ANIMATION, COMEDY, DOCUMENTARY, DRAMA, MUSICAL, and THRILLER as target movie genres.

Simões et al. [72] performed genre classification using MFCC computed from the audio and other visual features with a Convolution Neural Network (CNN) classifier. The authors reported that the inclusion of audio features improved the performance of all genres, especially the COMEDY genre.

Tadimari et al. [73] predicted the initial success of movies using a multi-modal approach. The

authors computed audio features like ZCR, energy, F_0, harmonic noise ratio, and MFCC. Genre classification was performed using a linear-kernel SVM classifier with audio-visual features. The genre prediction information and other meta-data were used for the prediction of movie success at the box office.

Wehrmann et al. [74] proposed an ensemble of 5 classifiers called *CoNNeCT* to perform movie genre classification. One of the classifiers in the ensemble was a Multi-Layer Perceptron trained on MFCCs computed from the audio. In later works, Wehrmann et al. [20, 75] performed multi-label genre classification using a deep residual CNN classifier that employs convolutions through time. The authors observed that the detection of HORROR genre was significantly improved with the addition of audio information.

Ben et al. [76] computed mid-level deep audio-visual features for genre classification. The authors used the ResNet-152 network [77] for extracting visual features and the SoundNet [78] for audio embeddings. The extracted features were trained with a probabilistic linear-kernel SVM classifier to generate a probability distribution over five genres. The genre probability distribution was used as a feature for movie interestingness prediction.

Álvarez et al. [79] performed aesthetic style clustering and genre classification of movies. The authors used low-level visual features and an SVM classifier for the tasks. It was observed that the inclusion of audio features improves genre classification performance.

Cascante et al. [28] performed genre classification of movie trailers using multiple modalities. Text, video, audio, posters, and meta-data were utilized in the task. For the audio modality, log-scaled Mel power spectrograms computed from 30s audio chunks were stacked and passed to a Convolutional Recurrent Neural Network for classification. The various modalities were combined using score fusion to obtain the final prediction. The authors observed that the addition of audio modality significantly improved the performance.

Chu et al. [80] proposed a multi-label genre classification system using movie posters. The authors used joint learning of visual appearance and object detection information for the task. A CNN sub-network inspired by AlexNet [81] and the YOLO object detector model version 2 [82] were combined. The YOLO model performed better at detecting animals, thereby improving the accuracy of ANIMATION genre prediction. Wi et al. [19] computed style features from movie posters for genre classification. A Gram layer was proposed in a CNN that uses a Gram matrix to extract style features from

movie posters. The use of the Gram layer in standard CNN-based models like *ResNet* [77] was shown to improve performance. Shambharkar et al. [83] performed genre classification using 3-dimensional CNN over stacks of video frames.

Mangolin et al. [84] combined information from audio, video frames, posters, subtitles, and movie synopsis in a late fusion framework for multi-label genre classification of the movie trailers. The authors used Binary Relevance and Multi-Label kNN (ML-kNN) classifiers.

Yadav et al. [36] attempted to classify the genres of movie trailers from Indian cinema. The authors extracted facial frames from the movie trailers and classified them into various emotions. The emotional mapping of the facial frames was used as a feature for genre classification. The authors proposed an Inception-LSTM-based classification system.

Fish et al. [85] stated that hard genre labeling of movies may not be reliable since genre definitions keep evolving with cultural progress [86]. Moreover, genre labels tend to be inconsistent across the movie durations [87]. Hence, the authors performed the genre classification task with a weak-labeling approach. The authors proposed a multi-label context-gated genre classification method. Fine-grained semantic clusters between movie trailer sub-sequences were learned in a self-supervised manner not restricted by the limitations of genre categories. Their multi-modal approach included scene understanding, image content analysis, motion style detection, and audio. The multi-modal information was combined using a collaborative gating method [88, 89]. Authors learned self-supervised semantic clusters by maximizing the cosine similarity between sub-sequences of the same movie while increasing the distance among dissimilar pairs. Audio embeddings used in their method were obtained from a network designed according to the Visual Geometry Group (VGG) style network. The said network was trained for audio classification [90]. The obtained audio embeddings were temporally aggregated using another network with a Vector of Locally Aggregated Descriptors (VLAD) layer. This network was termed NetVLAD [91]. The authors observed that the audio modality showed better performance in detecting COMEDY and SPORTS genres.

Sharma et al. [92] used only the audio modality and followed a bag of audio words based approaches to classify movie trailers into six genres, viz. ACTION, ROMANCE, HORROR, SCIENCE-FICTION, and COMEDY. The authors computed a set of 68 standard temporal and spectral audio features for 5s audio chunks and used the K-means algorithm to learn representative clusters. Distances from the learned clusters and the raw features were utilized to perform the final classification. The authors

observed that the ACTION genre was best detected with their proposed method.

Vishwakarma et al. [93] performed genre classification of movie trailers by extracting high-level cognitive and affective information obtained from multiple modalities of visual images, dialogues, and movie meta-data. The authors claimed that the dialogues could be used to extract cognitive information about the descriptions of the corresponding scenes.

2.2 Need for audio-based genre classification

The auditory stream is a rich medium for provoking various emotions [38]. Some specific sounds and music are frequently used by movie editors to elicit certain emotional responses and to promote dramatic effects [23]. Audio features were found to improve the performance of HORROR-scene detection [23]. The audio features are very strongly related to affective scenes [15]. The VSD performance improved with audio information [46]. The director's intent and style are represented by the movie music [24]. The musical scores of high intensity movies (ACTION and HORROR genre) are appreciably different from more steady emotive movies (DRAMA and ROMANCE). It was shown that a correlation exists between the rankings of viewers' emotional experiences (provided by self-assessments) and those obtained by automatic methods from the audio-visual features [39]. The audio information also helps in movie genre classification task also [28, 72, 74–76, 79, 85].

The affective content is better characterized by audio than video [38]. Various works have observed that the MFCC feature can be useful in the genre classification of videos [94,95]. The prosodic features like speech energy, pitch, fundamental frequency, and spectral features like MFCC are most popular in video affect detection [37]. Zhang et al. [26] explored rhythm-based music features to study the affective content of music videos. Certain specific patterns of environmental sounds are used to induce specific emotions. Moncrieff et al. [96] used this information to distinguish between HORROR and non-HORROR movies.

Previous literature discussed in section 2.1.3 indicates that most works in movie genre classification have focussed on the visual modality. Audio modality has been mostly used as an additional channel of information. Arguably, the work of Sharma et al. [92] might be one of the few genre classification methods developed solely using the audio modality. Nevertheless, most researchers have made a common observation that the addition of audio improves classification performance. More specifically, audio helps in improving the detection of COMEDY [72,85], SPORTS [85] and HORROR [20] genres. Thus,

the audio modality needs to be studied in more detail. The upcoming section reviews the literature on speech and music detection.

2.3 Detection of speech and music

Speech, music, and sound effects are the common audio types in almost all movie scenes [97]. Audio type segmentation (or audio source separation) is the first step in audio feature computation. Subsequently, separate features can be computed for speech, music and environmental sounds [37]. Wang et at. [38] separated MUSIC from ENVIRONMENTAL SOUNDS using chroma and energy features with a SVM classifier. Speech emotion recognition is the popular form of feature computation from the speech segments [37]. More efficient SPEECH, MUSIC, and ENVIRONMENTAL SOUND classification techniques have not been employed for movie genre classification. More accurate audio-type probability sequences will help in better feature learning and classifier training for movie genre classification. Speech and music are the most significant components of movie audio. This thesis attempts to develop state-of-the-art Speech vs. Music Classification (SMC) algorithms to work as preprocessing steps in the final genre classification system. The upcoming subsection reviews the literature on isolated speech and music detection.

2.3.1 Isolated speech and music classification

One of the first works in SMC was done by Saunders [98] for automatic real-time FM radio monitoring. Short-time energy and statistical parameters of the ZCR were used as features, and the categorization was performed using Gaussian Mixture Model (GMM) classifier.

Scheirer et al. [99] performed SMC as a preprocessing step for automatic speech recognition. The authors computed variance over 1 s segments for features like 4 Hz modulation energy, percentage of low energy frames, spectral roll-off, spectral centroid, spectral flux, ZCR, cepstrum features, and rhythm features. GMM, kNN, K-Dimensional trees, and Multidimensional Gaussian Maximum A-Posteriori (MAP) estimator were employed as classifiers. William et al. [100] performed segmentation of SPEECH vs. NON-SPEECH for an automatic speech recognition task. As features, mean entropy per frame, average probability dynamism, background-label energy ratio, and phone distribution match were used. The features were derived using posterior probabilities of phones obtained from the hybrid connectionist Hidden Markov Model (HMM) framework. The classification was performed using the Gaussian likelihood ratio test.

Zhang et al. [101] performed audio segmentation and retrieval for use in video scene classification, indexing of raw audio-visual recordings, and database browsing. The authors used features based on short-time energy, average ZCR, and short-time fundamental frequency. The classification was performed in two stages. A rule-based heuristic procedure for identifying SPEECH, MUSIC, ENVIRONMENTAL SOUNDS, and SILENCE was performed in the first coarse classification stage. In the second stage, an HMM classifier was used to classify the environmental sounds into finer classes.

El-Maleh et al. [102] targetted automatic coding and content-based audio or video retrieval applications with their SMC method. The authors used Line Spectral Frequencies (LSF), differential LSF, and measures based on the ZCR of the high-pass filtered signal as features. The kNN and Quadratic Gaussian classifiers were used for the classification task.

Moreno et al. [103] performed audio classification into SPEECH, MUSIC, and OTHERS. The target application of this work was automatic summarization of audio content. The authors computed the Fisher score over 26-dimentional MFCC and Δ MFCC feature vectors per frame. The score vectors were used with an SVM classifier for discrimination. Zhang et al. [104] performed automatic segmentation and classification of the audio signal into SPEECH, MUSIC, SONG, ENVIRONMENTAL SOUND, SPEECH WITH BACKGROUND MUSIC, ENVIRONMENTAL SOUND WITH BACKGROUND MUSIC, and SILENCE. This work used energy, ZCR, F_0, and spectral peak tracks as features. The classification was performed by using predefined rules. The authors detected harmonic and stable segments and then classified them into specific categories.

Lu et al. [105] aimed for audio content analysis in video parsing applications. The high ZCR ratio, low short-time energy ratio, Linear Spectral Pairs (LSP), band periodicity, and noise-frame ratio were used as features. This work employed a three-step classification approach. In the first step, kNN and Vector Quantization (VQ) of LSP were used for SPEECH vs. NON-SPEECH discrimination. The second step used heuristic rules to classify NON-SPEECH into MUSIC, BACKGROUND NOISE, and SILENCE. In the third step, speech regions were used for speaker segmentation. Bugatti et al. [106] performed SMC for generating a list of contents for a multimedia document. The authors used ZCR-based features, spectral flux, short-time energy, cepstrum coefficients, spectral centroids, the ratio of the high-frequency power spectrum, and a measure based on syllabic frequency in the task. Multivariate Gaussian classifier and a Multi-Layer Perceptron (MLP) were used as classifiers.

Pinquier et al. [107] performed SMC for describing and indexing an audio document. The 4 Hz

modulation energy, entropy modulation, number of segments, and segment duration were used as features. The authors used Naive-Bayes and Gaussian Inverse law based classifiers. Classifiers were trained on each feature separately. The final decision was made with the agreement of classifiers of 4 Hz modulation energy and entropy modulation. Any conflict was resolved using the classifier trained on the segment information. In another work, Pinquirer et al. [108] tried to combine speech-oriented and music-oriented classification frameworks. The speech was represented using 8 MFCC coefficients, energy and their derivatives, 4 Hz modulation energy, and entropy modulation. Music was represented using 28-Mel filter outputs, energy, number of segments, and segment duration. The authors used 128-component VQ initialized GMM classifiers for the task. Two separate classifiers were trained, one for SPEECH vs. NON-SPEECH and the other for MUSIC vs. NON-MUSIC.

Ajmera et al. [109] performed speech-music segmentation for automatic transcription of broadcast news. The authors first trained an MLP with the first 13 cepstra of a 12^{th}-order Perceptual Linear Prediction (PLP) filter to emit posterior probabilities of phones. Subsequently, a two-state HMM with minimum duration constraints was used to perform the segmentation. Averaged entropy measure and dynamism estimated at the trained MLP output were used as input features for the HMM.

Burred et al. [110] first performed audio classification into SPEECH, MUSIC, and BACKGROUND NOISE. Subsequently, music was classified into different genres. The ZCR, spectral centroid, spectral roll-off, spectral flux, spectral centroid, spectral flatness, first 5 MFCCs, harmonic ratio, beat strength, rhythmic regularity, RMS energy, time envelope, low energy rate, loudness, and a few others were used as features. The authors used kNN and a 3-component GMM based classifier. Alexandre-Cortizo et al. [111] proposed an SMC system that can work as a preprocessor to music genre classification systems. Time-frequency representation was computed using the Modified Discrete Cosine Transform and the Discrete Fourier Transform (DFT) with 512 sample frames. The authors used spectral centroid, spectral roll-off, ZCR, high ZCR ratio, short-time energy, low short-time energy ratio, MFCCs, voice-to-white ratio, and activity level from the computed spectrums. The mean and standard deviation of all the measures were computed over 43 frames as a feature for classification. The Fisher Linear Discriminant was shown to perform better than the kNN classifier.

Panagiotakis et al. [112] performed segmentation of an audio signal and SMC. Their work was motivated by the requirement of indexing and retrieval applications for audio-visual data. The authors performed segmentation using the RMS amplitude for 1 s segments with 50 sub-intervals of 20 ms.

The frames detected as change points were further checked for boundary point localization within a tolerance of 20ms. The authors used Matsushita distance as a dissimilarity measure. The SMC task was performed using a rule-based classification scheme using normalized RMS variance, probability of null Zero-Crossings (ZC), joint RMS-ZC measure, silent intervals frequency, and maximal mean frequency features.

Keum et al. [113, 114] performed SMC for speaker indexing. The spectrogram of each audio was non-linearly binarized using a predetermined threshold. The values greater than the threshold were set to one and otherwise to zero. Subsequently, the histograms of each frequency bin over 92ms blocks were computed and used as feature representation. The authors used a rule-based classification scheme in [113] and an MLP in [114].

Mesgarani et al. [115] were inspired by the auditory cortical processing model. The authors developed an SMC system for content-based audio classification in noisy and reverberant scenarios. Three-dimensional tensor-like features were extracted using Gabor-like spectro-temporal response fields. The authors reduced the data dimensions using higher-order Singular Value Decomposition and Principal Component Analysis. A Radial Basis Function (RBF) kernel SVM classifier was employed for the classification.

Muñoz-Exposito et al. [116] performed SMC for low bit-rate audio coding. The authors used a warped LPC-based spectral centroid feature in the task. Also, the spectral centroid, spectral roll-off, spectral flux, ZCR, and MFCC were employed as additional features. A fuzzy expert system based classification system was used and showed that it performed better than GMM classifier. Similar work was done by Garcia et al. [117]. Barbedo et al. [118] developed an SMC system for automatic segmentation in real-time applications. The authors used features based on ZCR, spectral roll-off, loudness, and fundamental frequencies. The kNN, Self-Organizing Maps, MLP, and their linear combinations were used for classification.

Farahania et al. [119] studied the effect of stationary additive noise in speech recognition. It was observed that high-pass filtering of the auto-correlation sequence of noisy speech could preserve significant spectral peaks. The auto-correlation of short-term frames of the speech signal was computed. This sequence was then high-pass filtered. A power spectrum was computed using FFT and hamming windowing of the filtered sequence. The power spectrum was then differentiated with respect to frequency. Finally, a Mel filter bank was applied to the differentiated signal to compute MFCC features.

The speech recognition task was performed using HMM classifier.

Song et al. [120] proposed an SMC system to improve the performance of the 3GPP2 Selectable Mode Vocoder. The spectral difference, music continuity counter, running averages each of 1^{st} LSF coefficient, energy, normalized pitch correlation, and periodicity counter were used as features. The GMM was used as a classifier. An energy-based Voice Activity Detection (VAD) was used at the first level to detect SILENCE and NON-SILENCE frames. Subsequently, the trained GMM classifier detected SPEECH and MUSIC from the non-silence frames. Multiple segments of SPEECH, MUSIC, and SILENCE were concatenated for the final decision.

Pikrakis et al. [121] segmented an audio stream and labeled each segment as either SPEECH or MUSIC. The proposed segmentation was performed in three phases. In the first pass, the region growing technique using the spectral entropy feature was used to identify speech and music segments. This method leaves out some audio segments as unclassified. Short-term energy, chroma vector-based features, and first 2 MFCCs were computed from the unclassified segments in the second phase. This phase was treated as a maximization task using a Bayesian Network and was solved using dynamic programming. In the last phase, a boundary correction algorithm was employed.

Seyerlehner et al., 2007 [122] proposed a novel feature termed *Continuous Frequency Activation* (CFA) for improved performance of music detection in Television broadcasts. The input audio stream was converted to mono-channel and was resampled at 11 kHz for the computation of CFA. A log-scaled power spectrum was further computed with a Hanning window of size 1024 samples and a shift of 256. The local spectral peaks were emphasized by using a 21 frame window size running mean. Subsequently, the power spectrogram was binarized using a predefined threshold. The mean activation of each frequency bin across 100 frames was computed with an overlap of 50 frames. Strong peaks in the activation function were detected, and their Height-to-Width Ratios (HWR) were computed. The peaks were then sorted based on their HWR, and the sum of the top five peaks was computed. This sum was considered as the measure of the overall peakiness of the activation function. The detection was performed by using a non-linear threshold. The authors observed that standard features and machine learning algorithms for SMC produced variable results, which were avoided by their CFA-based system. Lee et al. [123] performed detection of music in the sound-track of consumer-shot videos. The authors explained that sustained, steady musical pitches show significantly structured auto-correlation when calculated over hundreds of milliseconds. Also, it was noted that the auto-

correlation of aperiodic noise becomes negligible at higher-lag points after whitening a signal by LPC. Hence, the authors used auto-correlation based features with long-term averages. The GMM and SVM were used for classification. The authors observed that confusion occurred with weak or intermittent music, partial music, singing voice, and other high-energy sounds like screaming and alarms. Izumitani et al. [124] performed the detection of music overlapped by speech or other sounds in TV programs. The authors employed empirical features, MFCC, spectral powers of linear-scaled frequencies, and spectral power of Mel-scaled frequencies in the task. Principal Component Analysis based dimension reduction was performed. The authors used GMM and kNN classifiers. The accuracy was improved by frame-by-frame post-processing.

Anemüller et al. [125] detected the presence of SPEECH embedded in the natural acoustic background of NON-SPEECH sounds. The authors used an amplitude-modulated spectrogram computed in the "Meso-scale". Feature selection was performed, and SVM was used for the classification. It was observed that their method was slightly biased toward speech. Also, a degradation in performance for lower SNRs was observed. Taniguchi et al. [126] proposed to detect SPEECH, MUSICAL INSTRUMENTS, SINGING VOICE, and INSTRUMENTS WITH SINGING using spectral peak tracking. The sinusoidal trajectory of sounds was extracted, and 20 temporal features from the trajectories were computed. GMM was used for the classification. The authors observed that speech with background music was detected well using their method.

Lavner et al. [127] proposed a method for segmentation of audio signals into SPEECH and MUSIC. The motivation for this work was the development of consumer audio applications where real-time enhancements were applied. Authors used short-time energy, ZCR, skewness of ZCR, band energy ratio, low short-time energy ratio, autocorrelation coefficient, 10 MFCC, 10 Δ MFCCs, spectral roll-off point, spectral centroid, spectral flux, and spectrum spread as the features. Feature statistics were computed for segments of $2-6$ s. A Decision Tree was used for classification. The authors computed five thresholds to determine the types of an audio signal, viz., EXTREME SPEECH, EXTREME MUSIC, HIGH PROBABILITY SPEECH, HIGH PROBABILITY MUSIC, and SEPARATION (equal probability of speech and music). Gallardo-Antolin et al. [128] performed speech, music and song discrimination using MFCC Histogram Equalization [129] and Polynomial-fit Histogram Equalization (PHEQ) [130]. The PHEQ transformation contains the mean, variance, and shape of the distribution. The authors estimated the Cumulative Distribution Function using Order statistics-based equalization [129]. The

classification was performed using a GMM classifier.

Tardón et al. [131] proposed a SMC method. RMS, ZCR, cepstral residuals, spectral flux, MFCC, dynamic volume ratio, silence ratio, spectral centroid, spectral roll-off, spectral centroid, spectral bandwidth, frame energy, segment energy, F_0, and salience of pitch were used as features. Feature selection was performed to identify the most discriminative ones. The authors employed Fisher Linear Discriminant for the classification.

Bach et al. [132] employed perceptually motivated features for SPEECH detection in the presence of strongly varying external conditions like background noise. The authors used amplitude modulation spectral features computed from short-term spectrograms and SVM classifiers. BACKGROUND SOUNDS were detected in the first level, followed by SPEECH vs. NON-SPEECH classification. The authors concluded that the amplitude-modulated spectral features were more suitable for training robust speech vs. non-speech classifiers. Weninger et al. [133] performed speech separation from music by suppressing BACKGROUND MUSIC. The authors used the Non-negative Matrix Factorization method to estimate SPEECH and MUSIC bases in a semi-supervised manner. The background suppression was performed by using the estimated SPEECH and MUSIC bases. The authors observed that the semi-supervised method was effective in a highly noisy environment. However, a decreased overall quality of the enhanced speech was also observed at a high speech-to-music ratio.

Lim et al. [134] proposed an SMC system using running average of energy, reflection coefficients, partial residual energy, normalized pitch correlation, periodicity counter, along with music continuity counter as features. Separate feature sets were selected for SPEECH and MUSIC. The SVM output distribution was converted into conditional a-posteriori probability using the model-trust algorithm [135]. A logistic regressor trained based on the Bayes rule was used to smooth frame-wise predictions. The inter-frame correlation was utilized for refining the current frame's prediction.

Lim et al. [136] aimed at reducing expensive memory accesses and energy usage while maintaining efficiency in SVM-based SMC systems running on embedded devices. The authors proposed to reuse locally loaded support vectors to reduce memory access requirements. Such modifications increased the RBF-kernel-based SVM classifier's efficiency in running on the chip. Both improved performance and energy savings were observed.

Srinivas et al. [137] performed SMC to improve automatic transcription of the SPEECH signal, which is frequently interspersed with BACKGROUND NOISE or MUSIC in news videos. The authors

used 45 features per frame. This included 39-dimensional MFCCs, energy entropy block, short-time energy, ZCR, spectral roll-off, spectral centroid, and spectral flux. An Online Dictionary Learning (ODL) [138] based classifier was proposed for the task. The ODL was used to learn sparse dictionaries for each class. The L_1-Lasso algorithm was used to determine the dictionary representing the closest class to the test data during testing. The authors used 30 s segments for the classification. However, such large segments may be detrimental to the time resolution of classifier decisions in applications like continuous audio segmentation.

Neammalai et al. [139] developed an SMC algorithm with a focus on applications like efficient multimedia coding, automatic speech recognition, and automatic classification/indexing/archival/retrieval of information in large multimedia databases. The authors extracted texture information and energy signals from binarized spectrograms for use as features. The texture information was extracted from the spectrograms by zoning in the $[0-4]$ kHz spectral range and then computing local binary patterns. The energy signal was computed from the 2D Fourier transform of the binarized spectrograms. The authors compared SVM and MLP classifiers for the task and observed that SVM performed better.

Sell et al. [140] proposed two chroma-based features for the SMC task, viz., chroma difference and chroma high-frequency. The chroma vectors were computed using 12 bins, 100 ms frames with a shift of 25 ms. Additionally, existing features like normalized RMS standard deviation, silent interval ratio, silent interval frequency, ZCR variance, spectral centroid variance, Spectral flux variance, Mel-frequency subband modulation syllabic-rate energy, and Mel-frequency subband modulation spectral centroid were used. Feature statistics were computed over 1 s. The authors performed the classification using a GMM classifier.

Castán et al. [141] performed broadcast news audio segmentation using factor analysis. The audio stream was segmented into contiguous segments of SPEECH, MUSIC, SPEECH WITH MUSIC, SPEECH WITH NOISE, and OTHERS. The 16-dimensional MFCC with Δ and $\Delta\Delta$ blocks of overlapping 3 s intervals were used as features. The GMM Universal Background Model (UBM) super vectors were trained to compensate for within-class variability. The HMM backend was also used for the segmentation system. This method performed better than the GMM-based baseline.

Khonglah et al. [142] observed that speech is composed of different sound types like voice bars, high vowels, low vowels, and others. Such sounds have different Vocal Tract Constrictions (VTC) that differ from any music producing system. Thus, the authors proposed the use of variance of VTC

feature [143], chroma difference [140], and chroma high-frequency [140] as features that were computed over 30 s segments. Threshold-based non-linear mapping, GMM, and SVM classifiers were used.

Zhang et al. [144] proposed the use of i-vectors with Cosine Distance score and SVM classifiers for SMC. The authors argued that it was essential to filter out non-speech segments like MUSIC, NOISE, SILENCE, and OTHERS for better performance of applications like automatic speech recognition. Also, estimation of signal nature was essential for low bit-rate coders such as 3GPP2. The Universal Background Model was trained by using Expectation-Maximization algorithm [145]. Subsequently, the total variability matrix was trained, and i-vectors were extracted. Session compensation for i-vectors was performed using Within-Class Covariance Normalization and Linear Discriminant Analysis [146]. This work used 30 s audio segments for the training and testing phases.

Khonglah et al. [147] performed SMC using speech-specific features to extract information complementary to music signals. This work was motivated by the presence of more silence regions, shorter duration of vowels, and higher variation of the pitch in speech compared to music. The energy variance of the inverse of the 2^{nd} Mel-filter was used as the feature. Non-linear mapping and 2-component GMM with diagonal covariance matrices were used as classifiers. The authors also explored the efficacy of other speech-specific features for SMC in their next work [148]. The excitation source features like Normalized Autocorrelation Peak Strength of the Zero Frequency Filtered Signal, Peak-to-Sidelobe ratio of the Hilbert Envelope of the LP Residual were proposed for the SMC task. The vocal tract system features like Log Mel spectrum energy of the first 18 filters up to 2.5 kHz, and 4 Hz modulation energy were also included as features. In addition, existing features like ZCR, spectral centroid, spectral flux, spectral roll-off, and percentage of low energy frames were also employed. Thresholding-based non-linear mapping, GMM with diagonal covariance matrices, and RBF-kernel SVM classifiers were used. The statistics of features for 1 s segments were computed for training and testing.

Mezghani et al. [149] developed an SMC system as a preprocessing stage for automatic speech recognition and other tasks. The short-time energy, ZCR, the entropy of energy, spectral centroid, spectral flux, spectral roll-off, MFCCs, chroma vector, harmonic ratio, and fundamental frequency were used as features. The mid-term level statistics of the features for $1 - 10$ s segments were computed. The categorization was performed using the Decision tree, LDA, Naive Bayes, and SVM classifiers. The SVM was found to provide the best performance for the task.

Lopez-Otero et al. [150] performed audio segmentation for radio and television programs. The

authors mentioned that state-of-the-art audio segmentation systems fail in several real-world scenarios. Thus, this work proposed to fuse the results of standard methods in a weighted fashion for better performance. The class-conditional probabilities were estimated from the confusion matrices. These probabilities were used to perform a weighted decision-level fusion of an ensemble of segmentation strategies. The ensemble was found to perform better than the individual strategies. Vavrek et al. [151] performed hierarchical broadcast news audio classification into PURE SPEECH, SPEECH WITH MUSIC, SPEECH WITH ENVIRONMENTAL SOUND, PURE MUSIC, and ENVIRONMENTAL SOUND. The MFCC, the variance of Filter Bank Energies, the variance of $\Delta\Delta$ MFCC, band periodicity, spectral flux, spectral centroid, spectral spread, and spectral roll-off were employed as features. A Binary Decision Tree was used as a classifier. The authors observed the highest miss-classification error for MUSIC and ENVIRONMENTAL SOUND. In a previous work of the authors [152], they performed a similar task with temporal, spectral, cepstral, and pitch features and SVM binary trees. It was concluded that the binary tree-based scheme performed better than the one-against-one multi-class scheme.

Mohammed et al. [153] performed overlapped sound-tracks segmentation. The Singular Spectrum Analysis method was used to separate audio content into components with a reduced overlap of different classes. Subsequently, the components were categorized by using a random forest classifier. This method was found to perform better at 20dB and -20dB SNR.

Gimeno et al. [154] performed music detection with limited training data. Authors used Area Under ROC Curve (AUC) and Partial AUC optimization as loss functions with Recurrent Neural Networks. The authors explained that such methods overcome limited data problems and outperformed cross-entropy loss based models. Jia et al. [155] performed the joint task of MUSIC detection and MUSIC relative loudness estimation. It was observed that the joint task benefited from temporality and hierarchy. This helped in obtaining the solution. A hierarchical classification was performed. First, MUSIC VS. NON-MUSIC was classified, followed by FOREGROUND MUSIC VS. BACKGROUND MUSIC. A Hierarchical Regulated Iterative Network was proposed as the classifier for the task.

A summary of the various features, classifiers, and datasets used in the SMC literature is provided in Table 2.1. The features are grouped into eight categories, viz., temporal, spectral, cepstral, speech specific, music-specific, Linear Prediction (LP) based, image-based, and others. This grouping is done to understand the thrust area in the SMC literature. It can be observed (Table 2.1) that the temporal, spectral, and cepstral features are the most popular in this task.

Table 2.1: Summary of previous works in Speech vs. Music classification literature

Group	Features	Papers	Classifiers Used	Datasets [1]
Temporal	ZCR, skewness of ZCR, High ZCR Ratio, ZCR of the high-pass filtered signal, Band energy ratio, Low short-time energy ratio, Auto-correlation coefficient, percentage of low energy frames, Short-time energy, Entropy of energy, Normalized RMS standard deviation, Silent Interval Ratio and frequency, loudness, Voice to White, Activity level, the probability of Null Zero-Crossings, Joint RMS-ZC measure, time envelope, Pitch, Number of segments and segment duration [156]), noise-frame ratio (NFR)	[98–102, 105–108, 110–112, 118, 120, 121, 127, 134, 137, 140, 147, 149, 157, 158]	Heuristics based, GMM, Decision tree, LDA, Bayesian, SVM, Backpropagation NN, KNN, self-organizing maps, MLP	Radio France International (RFI) database, Scheirer & Slaney, GTZAN, EUSTACE, Broadcast News Database, TIMIT, MULTEXT corpus, MPEG-7 test data set
Spectral	Spectral centroid, Spectral flux, Spectral roll-off, Harmonic ratio, fundamental frequency, Spectrum spread, spectral difference, Maximal mean frequency, Spectral flatness, linear spectral pairs, energy in 4 sub-bands	[99, 101, 105, 106, 108, 110–112, 118, 120, 127, 137, 140, 148, 149, 157],	Decision tree, LDA, Bayesian, Heuristics based, SVM, Backpropagation NN, KNN, Self-organizing maps, MLP,	GTZAN, EUSTACE, Broadcast News, Scheirer and slaney, TIMIT, RFI database, MPEG-7 test data set,
Cepstral	Energy Variance of Inverse Mel Filter No. 2 (EVIMF2), MFCCs, Delta MFCCs, Mel-frequency subband modulation syllabic rate energy, PHEQ over MFCC, cepstrum-based feature	[106, 108, 110, 111, 121, 127, 128, 137, 140, 147, 149, 158],	Heuristics based, GMM, Decision tree, LDA, Bayesian, SVM, Backpropagation NN, KNN,	Scheirer & Slaney, GTZAN, EUSTACE, Broadcast News Database, RFI database,
Speech specific	NAPS, PSR, Log mel spectrum energy, 4-Hz modulation energy, VTC, syllabic frequency, phone distribution match, i-vector	[99, 107, 108, 142, 144, 148],	Cosine Distance Score (CDS), SVM, GMM, Heuristics based,	TIMIT, Broadcast News, Scheirer and slaney, GTZAN, Broadcast News, RFI database, MULTEXT corpus,
Music specific	Chroma vector, Chroma Difference, Chroma High Frequency, periodicity counter, music continuity counter, harmonic ratio, beat strength, rhythmic regularity, band periodicity	[120, 121, 134, 140, 142, 148, 149],	Heuristics based, GMM, SVM, Decision tree, LDA, Bayesian,	Broadcast News, Scheirer and slaney, GTZAN, TIMIT Dataset [159], commercial music CDS,
LP based	LPC, reflection coefficients, partial residual energy, Running average of 1st LSF coefficient computed from LPC, differential LSF, Warped LPC-based Spectral Centroid	[102, 109, 116, 117, 120, 134],	SVM, GMM, Fuzzy expert systems, HMM, KNN, Quadratic Gaussian classifier (QCG)	TIMIT Dataset [159], commercial music CDS,
Image based	Zoning spectrogram image, Texture feature extraction, 2D Fourier transform	[139]	SVM, MLP	–
Others	Auditory model output, MLP output, Posterior probability based Entropy and Dynamism features, Fisher score	[103, 115, 160],	SVM, HMM,	Speech(TIMIT), Animal vocalization (BBC Sound Effects audio CD collection), Music (RWC genre database), Environmental sounds (Noisex, and Auroa)

2.3.2 Spectral peak tracking

Spectral Peak Tracking (SPT) has been used in diverse signal processing domains. Audio signals can be considered to be composed of a linear combination of many sinusoidal basis functions. The SPT is an engineering approach to identifying a signal's most prominent sinusoidal components. This subsection reviews the previous works that have employed SPT in different applications.

McAulay et al. [161] proposed a sinusoidal modeling approach for analysis or synthesis system with applications to time-scale, pitch-scale modifications, and mid-rate speech coding. Component sine waves were obtained from speech signals by matching the peak amplitudes, frequency, and phases of adjacent frames in high-resolution Short-Term Fourier Transforms. Matched peaks across frames belong to continuing tracks. Sinusoidal components can represent these peak tracks. The authors mentioned that adding amplitude modulated sinusoidal components may be used to synthesize intelligible speech.

Smith et al. [162] also proposed a peak tracking algorithm to extract sinusoidal components for speech synthesis. The authors extracted individual sine wave components of a sound known as a partials. It was stated that the combination of multiple partials forms the timbre of a sound. The authors mentioned that the human ear is not sensitive to short-time phase distortions. Thus, identifying the essential partials from the magnitude spectrum might be sufficient to synthesize intelligible speech.

Parmanabhan [163] proposed a method for automatic speech recognition. The author used three bandpass filters in the range of the first three formant frequencies to filter the input signal. Subsequently, an IIR filter was used to track spectral peaks and their energies in each bandpass filtered signal. The author used these peak location and energy information with cepstral features for the classification.

Keum et al. [113] also proposed a form of spectral peak tracing for SMC. The spectrum of each audio frame was non-linearly binarized by using a predetermined threshold. Subsequently, the histogram for each frequency bin was computed and used as a measure to represent an audio interval.

Lagrange et al. [164] performed sinusoidal modeling of polyphonic sounds using partial tracking. A partial tracking algorithm was defined as the identification method of partial evolution over time. The earlier minimum frequency difference-based partial continuity identification was replaced with the linear prediction method.

Prerau et al. [165] attempted to characterize human Electroencephalography activity during anesthesia-

induced unconsciousness. The authors modeled a spectral peak with a Gaussian-like distribution. A particle filter based algorithm was used to track non-stationary components of EEG signal in the time-frequency domain. The parameters were initialized randomly and then updated iteratively.

Zhang et al. [166] performed heart-rate monitoring using spectral peak tracking. The highest spectral peak was first identified in a time window. A search region was designed to find at most 3 more peaks for the next time window. Subsequently, the detected peaks were verified using some heuristics. The verified peaks were used to form the peak track.

Murthy et al. [167] also used spectral peak tracking for heart-rate monitoring. The spectral peak tracking was initialized at every δ frame. The highest spectral peak in a frame was considered as the heart rate. Trajectories were formed by searching from that detected peak in the forward and backward directions. A measure called trajectory strength was defined for each unique trajectory, and the best one was selected. The selected trajectory was considered the heart-rate trajectory.

2.3.3 Overlapped speech and music detection

Speech and music are frequently found in overlapped conditions in real-world signals. Such signals need to be efficiently identified for further processing. The previous works have either attempted to separate the mixed components or have tried to enhance one component over the other. Such works mostly begin with the knowledge of signals having mixed components. However, not all audio signals in real-world scenarios consist of mixed signals. The standard enhancement or source separation methods might perform unpredictably in cases of clean speech and music signals. Therefore, methods to efficiently distinguish overlapping speech and music signals from their clean counterparts are essential. Nevertheless, this subsection reviews the literature on the separation and enhancement of speech with background music or noise.

Initial attempts toward signal separation and enhancement have been made using spectral subtraction-based methods [168–170]. Such methods have the disadvantage of introducing certain extra artifacts in the signal spectrum, which are referred to as musical noise. Many works proposed post-processing methods that can suppress musical noise, thereby enhancing the separated signals [171–173]. Suppression of musical noise has also been attempted using cepstral smoothing [174] and empirical mode decomposition [175]. Sørensen et al. [176] devised an algorithm for noise suppression that has different attenuation rules for speech-present and speech-absent regions. Explorations have also been made to improve upon the source separation methods using multi-channel inputs [177, 178]. Gannot

et al. [179] provided a comprehensible survey of different methods employed in multi-microphone and single-channel blind source separation. It was stated that both the research areas are on a converging path and borrow ideas from each other. A detailed survey of works in music source separation was done by Cano et al. [180].

Traditional methods of speech enhancement are highly effective in the case of stationary or slow-varying noise but not so much in the case of non-stationary noise [181]. Most of these signal characterization methods either have insufficient detail or tend to overfit specific noise instances [181]. Since only the noisy speech is available for processing, it pays to have apriori information about the signals involved. Some techniques have attempted to model the temporal dynamics of either speech [182], or corrupting noise [183] or both [184, 185] by using HMMs or linear dynamical systems. These techniques are employed to aid the localization of the current noise characteristics. Luo et al. [186] stated that the full range of vocal and instrumental dynamics are not captured by unsupervised methods like low-rank sparse modeling and others used in the literature. On the other hand, supervised methods like those based on NMF and others are slow and are not easily generalizable [186].

Most recent works have formulated the source separation problem in the deep learning framework. The works on deep learning-based source separation can be broadly grouped into three categories, depending on the interaction between input mixture and output targets [187]. These categories are Denoising-based approaches, Time-frequency mask-based approaches, and Multiple-target based approaches. Deep Recurrent Neural Networks (DRNN) have been successful in separating sources in single-channel acoustic mixtures [187]. Sebastian et al. [188] explored the Modified Group Delay feature [189] with DRNN to separate singing voice from music. Grais et al. [190] have harnessed the power of fully Convolutional Denoising Autoencoders. Two-dimensional spectro-temporal relationships for the single-channel source separation task were learned. Uhlich et al. [191] showed that data augmentation and blending of a feed-forward neural network (NN) and a recurrent NN model could improve the overall source separation performance. Yu et al. [192] tried to solve the cocktail party problem by separating multi-talker speech. A novel deep learning training criterion called Permutation Invariant Training (PIT) was proposed, which was different from the established multi-class regression technique and deep clustering methods. It was claimed that PIT directly minimized separation error. This technique also helped solve the problem of label ambiguity (or permutation). Kumar et al. [193] proposed an end-to-end clustering-based approach, Deep Attractor Network (DANet), to overcome

the in-class separation limitations. An Expectation-Maximization based algorithm was designed for DANet training. Muth et al. [194] proposed a deep network architecture that combines amplitude and phase features. It was shown that the phase derivatives were a good feature representation as opposed to the raw phase. Generative Adversarial Networks (GAN) have also been explored in the source separation task [195]. Typical deep network-based sound separation methods use synthetically mixed sounds while training the models. However, most available audio is not isolated. The human brain uses primitive cues (e.g., the direction of arrival, repetition, proximity in pitch and time, and others) to identify different sound scenes without the availability of large training datasets. Seetharaman et al. [196] proposed a bootstrapping-based, deep learning-based method for music source separation. The proposed method was built upon state-of-the-art deep clustering-based approaches. The need for ground truths was removed by training directly on primitive cues. A technique called primitive clustering was used. Such algorithms use soft time-frequency masks in the embedding space for clustering. High-confidence separated sources were remixed to augment the data.

Non-negative Matrix Factorization (NMF) based approaches were most popular in the field of source separation or signal enhancement [197, 198]. Raj et al. [181] aimed at the problem of speech enhancement in a mixture of SPEECH and MUSIC using the NMF technique. Exemplar-based methods were used to learn over-complete sets of NMF bases for the signals. Weninger et al. [133] stated that it was unknown whether the information carried by fully supervised NMF models (like [181]) was optimal for the signal separation or enhancement task. A semi-supervised method was proposed for suppressing background music using NMF that preserves NMF dimensionality and reduces computations. It was shown that the on-the-fly estimation of music bases performs better than a supervised explicit background music model. It was claimed that the proposed semi-supervised method could suppress music to a larger extent in highly noisy conditions than the supervised method. However, since the modeling technique has a lossy characteristic, an overall decrease in quality was observed in the case of higher speech-to-music ratios. Abdali et al. [199] targetted speech-music separation. According to the authors, considering each frame of a signal as an independent observation was a drawback of the standard NMF method. Also, fixing the same decomposition rank for all component signals may not be correct. The authors proposed to use a regularization term for the signal frames in the cost function to consider spectro-temporal continuity. Also, different decomposition ranks for speech and music (smaller in music than speech, considering more variations in music signals) were used.

Moreover, a filter to improve the signals estimated by the NMF method was constructed.

Huang et al. [187] stated that in real-world scenarios, linear models were not expressive enough to model the complicated relationship between separated and mixture signals. The authors explained that signals might not always follow Gaussian distributions. The mapping relationship between the mixture signals and separated sources was considered a non-linear transformation. Hence, it was argued that non-linear models such as deep neural networks (DNNs) were desirable. The authors targetted the monaural source separation problems like speech separation, singing voice separation, and speech denoising tasks. The authors explored various deep learning architectures. Joint optimization of a soft time-frequency mask layer was proposed with different network architectures to improve state-of-the-art performances. A discriminative objective function was explored to reduce signal interference and achieve better results. Deep clustering is a method of clustering interfering signals into different clusters in an embedding space. It has been a successful method in separating a mixture of speech signals [200]. Luo et al. [186] targetted vocals and music separation using the deep-clustering method. A two-headed Chimera network with both deep-clustering and mask-inference heads in the same network was proposed.

2.3.4 Datasets

The *Moviescope* dataset [28] is a recent and comparatively large movie trailers dataset available in the public domain. It consists of approximately 5000 movie trailers. This thesis uses the *Moviescope* dataset for developing the proposed approaches. Apart from the *Moviescope* dataset, various other datasets of varying sizes and structures are also created by other researchers. Some of them are proprietary, while many others are not active now. Nevertheless, all the significant datasets popularly used in the literature on movie genre classification are provided here as a reference.

- *VSD96* [55] consists of 96 hours of violent scene data.
- *MMX-Trailer-20* [201] consists of approximately 8000 trailers. The movies are assigned genre labels from a list of 20 different labels obtained from IMDb. Each trailer is assigned at most 6 genre labels.
- Emotion-based Genre Detection for Bollywood (*EmoGDB*) [36] dataset consist of 100 Indian movie trailers, annotated with one unique genre label out of six possible genres.

- *Moviescope* [28] consists of 5000 trailers with at most 3 genre labels for each trailer out of 13 possible labels.
- *MMTF-14K* dataset [202] is composed of approximately 14000 trailers and labeled from a list of 18 genre labels.
- *Movie Trailers Dataset* [73] consists of approximately 500 trailers with genre assignments form a list of 10 genre labels.
- *EmoGDB* [36] is a small dataset consisting of 100 Hindi movie trailers. Each trailer has a single genre label out of six possible ones.

For the speech and music classification tasks, the *MUSAN* dataset [203] is the most popular one. Apart from the *MUSAN* dataset, this thesis also uses the *GTZAN* [204], *Scheirer-Slaney* [99], *Muspeak* [205] and the *DAFx-12* [206] datasets for validating the performance of the proposed algorithms. These datasets are further discussed in the subsequent chapters.

2.4 Organization of the work

The plan for this thesis is developed based on the review of existing literature. This section highlights the gaps in the present literature and accordingly identifies the possible directions for further exploration.

The magnitude-spectrum based features in the SMC literature model only the gross tempo-spectral characteristics of SPEECH and MUSIC. Very few works have tried to model the unique striation patterns of these signals. Isolated SPEECH or MUSIC signals can be detected from the striation patterns in their spectrograms. This is evident from their visualization. For machine listening systems, new features are required that can model the distinct characteristics of the two audio categories. This motivated us to develop a novel feature extraction methodology from the striation patterns of speech-music spectrograms. This is described in chapter 3 of this thesis.

The current literature in SMC lacks a detailed study on the phase characteristics of speech and music signals for their discrimination. The significantly different methods of generating these audio signals must affect their phase characteristics. This aspect is analyzed in detail in chapter 4 of this thesis. Three existing phase-based features are explored for the SMC task. Their efficacy in capturing the discriminating information is studied in the chapter.

The challenging situation of speech overlapping with music (SPEECH+MUSIC) has been tackled in previous literature, mainly for source separation and signal enhancement applications. Most of such approaches either have apriori signal information or make assumptions on mixing proportions. However, such information may not be available in real-time processing scenarios. Automatic methods for detecting such overlaps are necessary. Some researchers have previously tried to classify various audio types, including SPEECH+MUSIC signals. However, such methods consider SPEECH+MUSIC as an independent category with no relation to clean SPEECH or MUSIC signals. This thesis considers SPEECH+MUSIC as a noisy version of either clean SPEECH or clean MUSIC signals. In this respect, previously unexplored Harmonic-Percussive Source Separation based features and Multi-Task Learning based classifiers are explored to detect speech+music in chapter 5.

The available literature on movie genre prediction is significantly biased towards the use of visual information for the task. The audio modality has been mostly used as supplementary information. However, the richness of movie audio modality demands special attention to it. Hence, this thesis attempts to perform genre prediction of movies using only the audio modality. We believe that the SPEECH and MUSIC information of the movie audio might relate to its genre. Thus, SPEECH and MUSIC detection methods developed in this thesis are employed in the task of movie genre prediction in chapter 6. The proposed algorithms are validated on datasets of movie trailers that represent a concise form of a movie.

The next chapter describes the proposed method for SMC using novel tempo-spectral features. A novel SPT method is proposed and is used to compute the features. The proposed features are observed to capture the unique striation patterns of speech and music signals.

3

Magnitude Features for Speech Music Classification

Publications

- **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Speech/Music Classification Using Features From Spectral Peaks," in *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 28, pp. 1549-1559, 2020.

Contents

Objective

Movie audio is composed of significant amounts of speech and music signals. Efficient processing of movie audio necessitates a detailed study of its constituent signals. This chapter attempts to understand the discriminating characteristics of speech and music. The time-frequency properties of audio signals can be analyzed using two-dimensional representations like spectrograms. Magnitude spectrograms of speech and music contain distinct striation patterns. Traditional audio features represent various audio signal properties but do not necessarily capture such patterns. Therefore, a novel Spectral Peak Tracking (SPT) approach is proposed in this chapter to model such patterns in the spectrograms of speech and music. The chapter further proposes two novel tempo-spectral features for speech vs. music classification. The proposed features are extracted in two stages. First, SPT is performed to track a predefined number of highest amplitude spectral peaks in an audio interval. In the second stage, the location and amplitude information of these peak traces are used to compute the proposed feature sets. The first feature involves the computation of the mean and standard deviation of peak traces over an audio interval. The second feature is the averaged component posterior probability vectors of Gaussian mixture models learned on the peak traces. Speech vs. music classification is performed by training various binary classifiers on these proposed features. Three standard speech-music datasets are employed for evaluating the proposed features. In addition to the standard datasets, a Movie-MUSNOMIX *dataset is created from the audio of four Indian movies that consists of annotations for seven audio categories, including speech and music. Performances of the proposed features are also assessed on the* Movie-MUSNOMIX *dataset to establish the usefulness of this study in analyzing the primary target signals of this thesis. The proposed features are benchmarked against five baseline approaches. Furthermore, the best-performing proposed feature is combined with two contemporary deep-learning-based features to show that such combinations can lead to more robust speech vs. music classification systems. Finally, the proposed features are also evaluated on real-world speech and music signals from the* DAFx-12 *and* Muspeak *datasets to analyze their generalizability to continuous audio sequences consisting of natural transitions between categories and possible overlaps.*

3.1 Task overview

Content-based indexing and retrieval applications often involve a critical preprocessing step of segmenting and classifying the movie audio modality into distinct categories. Such applications require

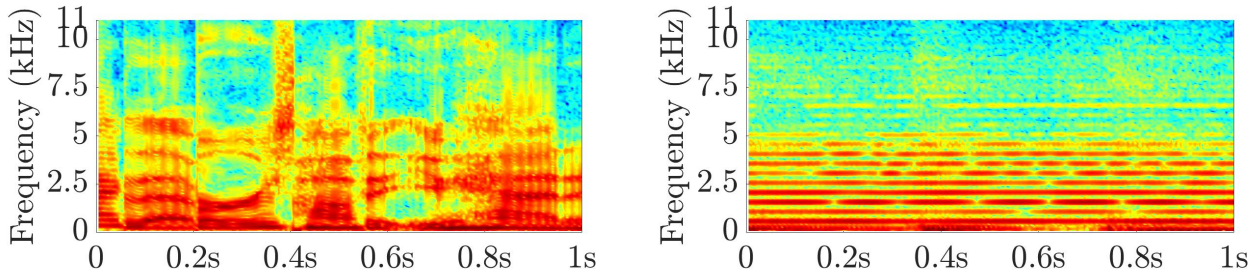


Figure 3.1: Spectrograms of (a) Speech and (b) Music, computed using window size of 10 ms and frame size of 5 ms. It may be noted that speech and music display distinct striation patterns in their respective spectrograms. This observation motivated the proposal of time-frequency audio features for speech-music discrimination described later in this chapter.

efficient classification algorithms that ensure homogeneity of individual audio categories in the detected segments [207]. Speech and music are the most frequently encountered audio categories in nearly all non-silent movies [97]. Thus, efficient classification of these signals is vital for audio segmentation-based applications. Apropos of such a necessity, this chapter proposes novel feature representations for better Speech vs. Music Classification (SMC).

Researchers have exploited various acoustic differences between speech and music signals for classifying them [127,208]. Saunders et al. [98] mentioned that pitch information usually exists for only three octaves in speech, whereas fundamental tones in music span up to six octaves. Sell et al. [140] stated that, unlike speech, music is expected to have strict structures in the frequency domain since specific tones play an essential part in its production. Panagiotakis et al. [112] showed that the amount of silence present in the signal might also be a good discriminator between the two classes. Short silences usually punctuate speech sound units, while music is generally continuous (Fig. 3.1).

Many standard audio features have been used in literature to model the distinct behaviors of speech and music. The most widely used spectral features in this task are Zero-Crossing Rate [140], Spectral Centroid, Spectral Roll-off, and Spectral Flux [149]. Energy [139], Entropy [137] and Root Mean Square [140] values are the most popular temporal features used in SMC. Khonglah et al. [148] have proposed that features predominantly used in speech processing tasks (like the speech-specific modulation spectrum features) can be effective in the current task also. On the other hand, Sell et al. [140] suggest that chroma-based features are better in modeling the octave patterns in music and thus might be useful in discriminating it from speech.

Existing works in SMC have mostly employed Gaussian Mixture Models (GMM) [140, 144, 148], Artificial Neural Networks [137], k-Nearest Neighbors [110, 111, 118] and Support Vector Machines (SVM) [139, 144, 149] as classifiers. Recently, researchers have also used deep learning techniques for solving the SMC task [209, 210]. Convolutional Neural Network (CNN) is very popular in image processing applications for feature learning and classification. This motivated researchers to use CNNs to learn features from audio segments' time-frequency representation. For example, Doukhan et al. [211] proposed a CNN-based semi-supervised training procedure for solving the SMC task. The first convolution layer in their architecture is pre-trained unsupervised using the spherical k-means algorithm and later kept constant throughout the model training phase. The input to their model is the stacked Mel-frequency coefficients of 50 frames. In contrast, Papakostas et al. [212] used transfer learning to fine-tune an existing CNN model trained on an image classification task to learn the discrimination between spectrograms of speech and music.

There are works in the literature that have explored features that can capture simultaneous variations in temporal and spectral domains for achieving better performance in SMC [134,137,140,148,149]. Spectrograms are a popular method of visualizing the tempo-spectral properties of an audio signal. Fig. 3.1(a) and Fig. 3.1(b) show the spectrograms of speech and music respectively. It may be noted that spectrograms can either have high time-resolution (wideband spectrograms) or high frequency-resolution (narrowband spectrograms), but not both at the same time [213]. Wideband spectrograms are generated with short temporal windows. They are characterized by vertical striations representing pitch period and formant frequencies (in the case of speech) in the form of prominent horizontal bands [213]. Narrowband spectrograms are generated using longer analysis windows and horizontal striations showing the fundamental frequency and its harmonics. Peaks in the spectra of audio frames may appear as striation patterns in spectrograms. Distinct class-specific properties can be captured by tracing trajectories of the highest spectral peaks in the spectrograms. Researchers have also used spectrograms for feature extraction. For example, Mesgarani et al. [115] were inspired by the auditory cortical processing methods to use Gabor-like spectro-temporal response fields for feature extraction from spectrograms. Whereas Neammalai et al. [139] performed thresholding and smoothing on standard spectrograms to form binary images and used them as features for classification.

Peak tracking has been a widely explored approach in speech coding and synthesis. McAulay et al. [161] proposed that a speech segment can be represented as a combination of various sinusoids of

specific frequency and definite lifetime, called partials. They generated high-quality artificial speech by adding different partials with time weighing and amplitude modulation. Smith et al. [162] proposed a similar approach to [161], but for representing polyphonic music. Lagrange et al. [164] proposed an improved partial tracking algorithm based on the linear prediction algorithm that can better model the pseudo periodic part of polyphonic sounds. In other works, researchers have used a technique called Spectral Peak Tracking (SPT) to trace the trajectory of fundamental frequency across consecutive frames in the spectrogram [166, 167]. Techniques similar to SPT have also been used for feature generation in SMC literature. Seyerlehner et al. [122] proposed a feature called Continuous Frequency Activation (CFA), which measures the steadiness of spectral components within a block of audio. Since music is considered relatively more stationary, this feature provided improved results in the case of music detection. In works like [113], authors used a pre-determined threshold to binarize the magnitude spectrum of each audio frame to ones and zeros within an interval. Subsequently, they counted the number of ones that appear for each frequency channel and used this measure as a feature for classification. Padmanabhan et al. [163] processed speech signals using band-pass filters and tracked spectral peaks in each band for speech recognition.

It can be observed from Fig. 3.1 that speech and music signals produce quite distinct striation patterns in their respective spectrograms. Pitch and harmonics in speech slowly change from one sound unit to another [214]. These gradual transitions create arc-like striations in speech spectrograms. On the other hand, the music signal consists of relatively stationary pitch and harmonics with sharp transitions [215, 216]. Such characteristics appear in music spectrograms as horizontal line segments with sudden breaks. These spectro-temporal differences observed in the spectrograms of speech and music can be attributed to the following reasons. First, the speech production system possesses inertia [217, 218]. It requires a relatively large amount of time to change from one sound unit to another, leading to a smooth transition between sound units in speech spectrograms. On the other hand, individual notes of music have a specific onset instant, marked by a relatively large burst of energy that makes its striation patterns discontinuous [219]. Second, music tones decay slowly [220]. In contrast, speech production is a damped system where sound units decay quite fast [221]. This phenomenon explains the horizontal patterns in music but not in speech. Third, musical instruments produce only a fixed number of tones and their overtones [140]. On the other hand, the speech production system generates a large number of intermediate frequencies while transitioning from one

sound unit to another [147]. Such an occurrence leads to the formation of arc-like patterns in speech spectrograms. Nevertheless, horizontal striations in music may be absent in the case of some instruments and playing styles. For instance, many melodic instruments have continuously varying pitch. Additionally, orchestral movie theme music often lacks strong percussive components because of only multiple melodic instruments playing together. Similarly, speech shows percussive striations for plosive bursts and discontinuities at many consonant-vowel boundaries. However, the observation that music consists of more horizontal and vertical striations than speech may be assumed to hold in a generic sense.

The observed differences in the spectrograms of speech and music motivated us to design features that can capture these distinct class-specific striation patterns for speech vs. music classification. However, hand-crafted features have a high dependence on problem-specific assumptions. On the other hand, automatic feature learning methods (like CNNs) can efficiently learn underlying patterns in the data. However, it is not easy to interpret the information learned by such deep-learning methods due to their inherent stochastic nature. In applications where domain knowledge is available, it is also worthwhile to explore hand-crafted features that can provide decent performance. Efficient hand-crafted features may be combined with deep-learning-based features to build more robust systems for SMC. These ideas form the basis of the current proposal. Accordingly, this chapter has the following contributions.

(i) A novel approach for SPT capable of capturing prominent striation patterns present in spectrograms of speech and music signals is proposed (section 3.3.1).

(ii) Two novel feature sets are proposed. The MSD features are constructed using first and second-order moments of location and amplitude values of peak traces obtained by SPT (subsection 3.3.2). Second, CBoW features are constructed using averaged posterior probabilities computed from Gaussian mixture models learned on peak traces obtained from entire training data (section 3.3.3).

(iii) The proposed features are combined with learned features extracted using contemporary deep networks. Experimental results show that such combinations can build more robust SMC systems.

(iv) A *Movie-MUSNOMIX* dataset is created from four Indian movies with annotations for seven

different audio categories, including speech and music. This dataset is created with two aims. First, to evaluate the proposed algorithms on the primary target signals of this thesis. Second, as a contribution to the community for facilitating the study of the complexities of Indian movie audio.

The rest of this chapter is organized as follows. Detailed descriptions of the proposed scheme for SPT (section 3.3.1) and subsequent feature extraction procedures (sections 3.3.2, 3.3.3) are provided in Section 3.3. The details related to the construction of *Movie-MUSNOMIX* dataset are described in section 3.2. The design of the experiments and their results are presented in section 3.4. Finally, the chapter is summarized in section 3.5.

3.2 Creation of *Movie-MUSNOMIX* dataset

The primary goal of this thesis is to perform genre classification of movies. Various studies are performed in this thesis to analyze and classify the audio content into various homogenous categories. For instance, the current chapter deals with proposing novel features for speech vs. music classification. Different standard audio datasets (available in the public domain) are used to develop the proposals of this thesis. Nevertheless, to gauge the effectiveness of proposed algorithms in the target domain, it is pertinent to evaluate them on real movie audio. However, to the best of our knowledge, standard movie audio datasets with annotations for various component audio categories are not available in the public domain. This lacuna puts the onus on us to create such a dataset to serve the purpose of this thesis. Hence, this thesis contributes a new dataset to the community that consists of seven audio classes. The dataset is named as Movie-*MU*sic, *S*peech, *NO*ise and *MIX*ed (*Movie-MUSNOMIX* [1]). The *Movie-MUSNOMIX* dataset consists of approximately 8 hours and 20 minutes of audio content in total. It consists of audio extracted from four Hindi movies (released from 1966 to 2011). Audio segments obtained from the movies have been manually annotated into one of the following seven categories – SPEECH (Sp), MUSIC (Mu), NOISE (No), SPEECH+NOISE (SpNo), SPEECH+MUSIC (SpMu), MUSIC+NOISE (MuNo) and SPEECH+MUSIC+NOISE (SpMuNo). The annotation procedure employed for creating this dataset is described next.

Table 3.1: A summary table describing the *Movie-MUSNOMIX* dataset.

	Sp	Mu	No	SpMu	SpNo	MuNo	SpMuNo
Duration	1Hr 48Min	1Hr 51Min	24Min	2Hr 49Min	51Min	18Min	15Min
Number of files	203	307	117	263	139	78	54
Minimum file duration	≈ 1 s	≈ 1 s	≈ 1 s	≈ 1 s	≈ 2 s	≈ 1 s	≈ 1 s
Maximum file duration	≈ 105 s	≈ 100 s	≈ 71 s	≈ 110 s	≈ 92 s	≈ 50 s	≈ 123 s
Average file durations ($\mu \pm \sigma$)	32.19 $\pm$27 s	21.84 $\pm$18.86 s	12.56 $\pm$12.41	38.66 $\pm$32.09 s	17.49 $\pm$18.74 s	14.06 $\pm$12.5 s	22.27 $\pm$25.15 s
Storage format	Variable bitrate mode mp3 ($\approx$ 192Kbps)						

3.2.1 Annotation procedure

The *Movie-MUSNOMIX* dataset is created as a collection of folders, one each for the seven audio categories. The movie audio sequences are split into short chunks and annotated as a single audio category. Individual segments are stored in specific folders according to their respective labels. The annotations have been made using the Praat software [222]. One annotator performed the initial annotations, followed by an independent verification by a second annotator. Both the annotators were Ph.D. scholars at IIT Guwahati researching in the field of speech processing. The annotators used over-the-ear Corsair HS50 stereo gaming headphones while annotating. Following guidelines have been followed during the annotation process.

(i) Foreground sounds have been treated as either single (Sp, Mu, or No) or composite (SpNo, SpMu, MuNo, or SpMuNo).

(ii) An audio file is labeled as the prominent foreground sound class (single or composite) based on the perception of the annotator.

(iii) Background sounds present in an audio file (if any) have been noted with their onset and offset times.

(iv) The song sequences consisting of both vocal and instrumental music have been marked as Mu.

[1]Available online at URL: `https://github.com/mrinmoy-iitg/Movie-MUSNOMIX`

The audio segments are stored in the *mp*3 format with variable bitrate mode (approximately 192Kbps). The choice of storing the audio segments in *mp*3 format was made due to its low memory footprint. Moreover, Urbano et al. [223] showed that standard features like Mel-Frequency Cepstral Coefficients (MFCC) and Chroma are robust to *mp3* encoding with a minimum bitrate of approximately 160Kbps. Also, *mp3* compressed files with high enough bitrate ($>$ 160Kbps) are shown to retain sufficient information even to recover phase-coding based hidden data in audio signals with near-zero error rates [224]. A summary of the dataset information is provided in Table 3.1.

3.2.2 Analysis of the corpora

An analysis of the *Movie-MUSNOMIX* dataset provides insights into the structure of Indian movie audio signals. The duration of the speech category in the dataset is approximately 1 hour 48 minutes. Music has a duration of approximately 1 hour 51 minutes. Approximately 24 minutes of the dataset consists of environmental sounds and noises. Speech with noise has a duration of approximately 51 minutes, whereas music with noise is present only for around 18 minutes. Around 15 minutes of the dataset is composed of speech with music with noise. Almost one-third of the *Movie-MUSNOMIX* dataset (approximately 2 hours 49 minutes) is composed of speech with music. This imbalance in the sizes of different classes might be evident from the principles of sound editing employed in Indian movies.

The audio content of Indian movies reflects the significant influence of the Indian cultural context in terms of spoken language, music genres, vocal songs, and others. Indian movies frequently use different kinds of background music to highlight the moods of various dramatic scenes. This trend is indicated by the large size of the SpMu class. The genres of music generally present in Indian movies are traditional folk, ghazals, Hindustani or Carnatic classical and semi-classical, fusion with western music, cabaret, and others. Choudhury et al. [225] noted that there had been an increasing influence of western and middle-eastern music in contemporary Indian movie music. Mukherjee [226] provided a detailed and insightful study into the development of Indian movie music over the years. Indian movie music consists of traditional instruments like bamboo flute, shehnai, tabla, sitar, and harmonium, in addition to pianos, electric guitars, drum sets, and many other western musical instruments. Thus, the soundscape of Indian movie music is generally vibrant and complex. It often consists of large orchestras where multiple musical instruments are played simultaneously. Vocal songs are typically created with the singing voice of one or two lead singers, accompanied by a chorus and instrumental music. The Indian

movie audio often consists of ambient noises from the surrounding environment where a particular scene is portrayed, like a street, crowd, indoor or outdoor, and animal sounds. Unfortunately, Indian movie audio is largely underexplored. Therefore, the *Movie-MUSNOMIX* dataset is a small contribution by the authors to encourage audio analysis and related research in the context of Indian movies. The following section describes the proposal of computing novel audio features for SMC in detail.

3.3 Proposed work

Speech and music are complex non-stationary signals. Spectra of speech consist of source harmonics superimposed by vocal tract formants. Energy concentrations in the spectrograms of speech signals are a manifestation of formants [227]. High-amplitude peaks in speech spectra form these energy concentrations. It has been established that high-amplitude peaks provide information about dominant formants in the speech spectra [163, 228]. It may be noted that formants are defined only for voiced segments, which constitute most of the speech content. In contrast, the music spectrum does not have any formant-like structure. It is composed of harmonics and resonances of the constituent instruments. High amplitude spectral peaks in the music spectra mostly correspond to resonant frequencies that influence the timbre of various instruments. Spectral peak trajectories carry valuable information about the underlying sound segment. Tracking the high amplitude peaks in speech and music might provide enough information to discriminate between these signals. However, the number of high amplitude spectral peaks to be considered for tracking might be a task-dependent tunable parameter. For example, many fundamental frequencies in polyphonic music or multi-speaker speech might be better represented with additional spectral peak tracks.

Most speech processing systems employ perceptually motivated cepstral features that do not explicitly model the peak trajectory information [163]. Performing SPT in the spectrograms of speech and music might be an effective way of extracting discriminating features because of three strong reasons. First, speech formants have a well-studied structure and show a predictable behavior [229]. However, resonances in music have a dynamic nature depending on the composition and instruments used to produce the signal. Second, speech production uses only a single resonant cavity, i.e., the vocal tract. In contrast, the music signal is composed of multiple resonant devices depending upon the number of instruments used. Any deviation from the resonance patterns of speech may indicate the presence of music in the current two-class scenario. Third, as discussed in Section 3.1, music signals

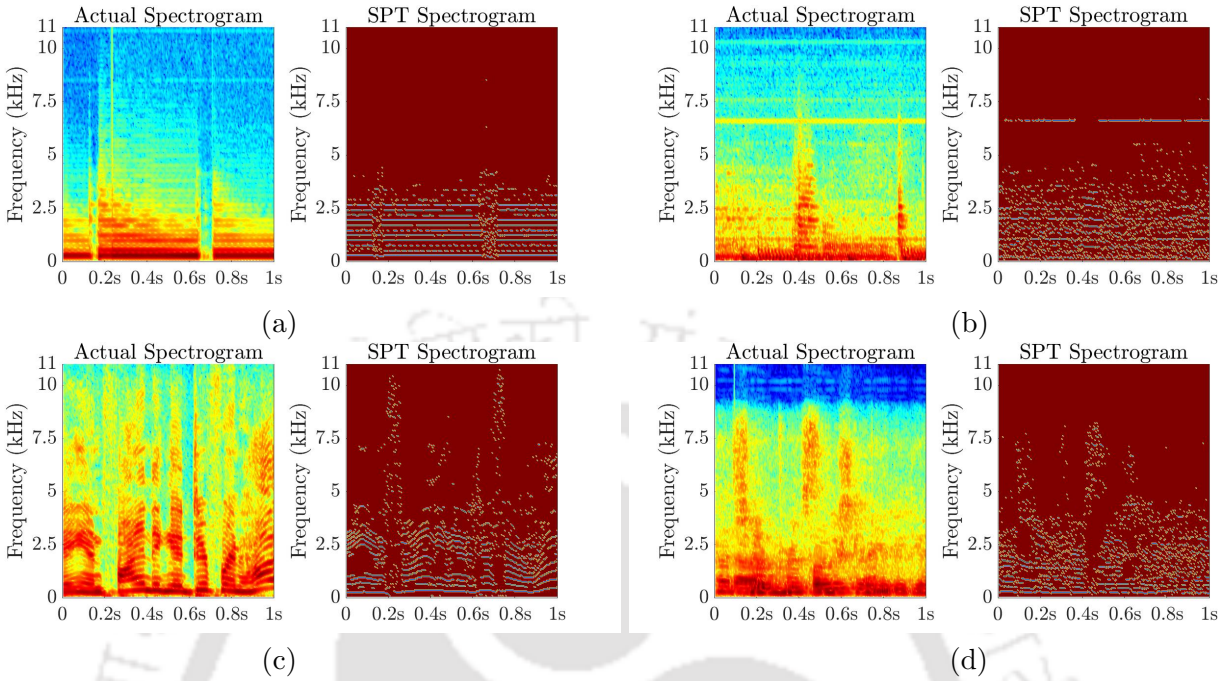


Figure 3.2: Illustrating the effect of multiple fundamental frequencies on the spectrograms and subsequently on the proposed peak tracking algorithm. The figures shown are spectrograms computed from 1s intervals of (a) monophonic music, (b) polyphonic music, (c) single speaker speech, and (d) multi-speaker speech. The original spectrogram is shown on the left, and the SPT spectrogram is shown on the right.

tend to maintain some of their spectral striations for a considerable duration of time, whereas speech is highly non-stationary. Therefore, the trajectories of spectral peaks extracted from spectrograms are believed to capture such distinct behavior of speech and music. The proposed SPT technique and the subsequent feature extraction procedure are described in the following subsections.

3.3.1 Proposed SPT method

This chapter proposes a novel SPT algorithm that is designed to capture a preset number of highest spectral peaks in the spectrogram frames of an audio signal. An audio segment $x[n]$ ($x[n] \in \mathbb{R}^1$) and $n = 0, \ldots (N_s - 1)$ is divided into L overlapping frames x_l ($l = 0, \ldots (L-1)$) of size $2N_f$. Let, the k^{th} DFT coefficient of x_l be

$$\mathbf{X}_l[k] = \sum_{m=0}^{(2N_f-1)} x_l[m] e^{-jk\frac{2\pi}{2N_f}m} \tag{3.1}$$

where, $k = 0 \ldots (2N_f - 1)$. These frames (x_l) are sequences of real numbers. Hence, only the first N_f DFT coefficients (i.e. $\mathbf{X}_l[k]$ where, $k = 0, \ldots (N_f - 1)$) are considered for further processing.

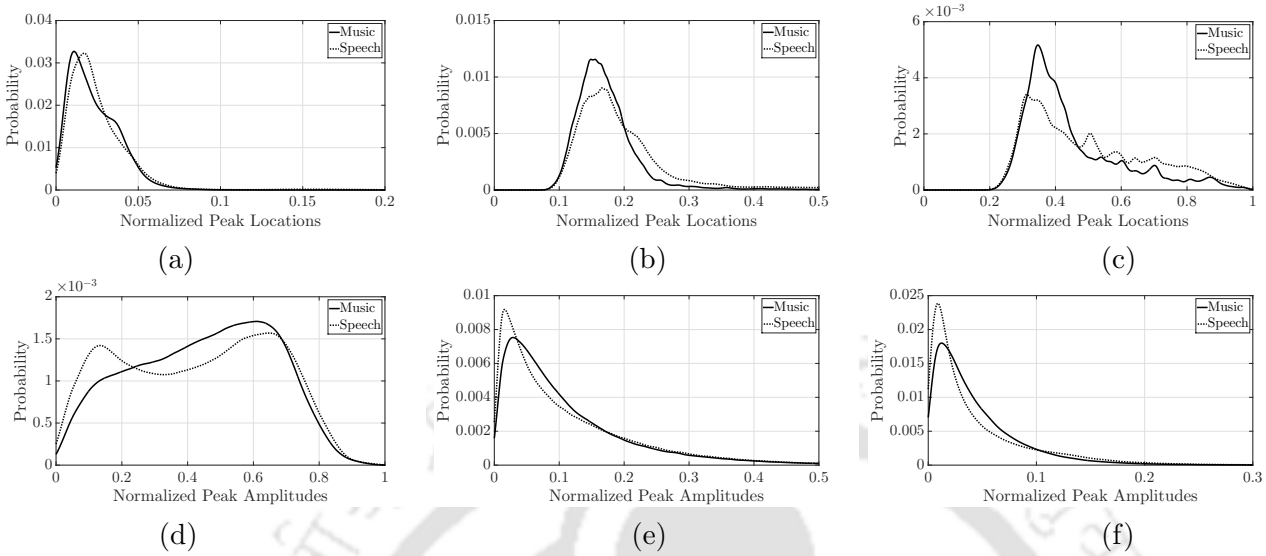


Figure 3.3: Illustration of the peak location and amplitude distributions for 1^{st}, 5^{th}, and 10^{th} peak traces, generated using window size of 10 ms and frame size of 5 ms. Figures **(a)-(c)** show peak location distributions and Figures **(d)-(f)** show peak amplitude distributions. The data is drawn from the *GTZAN* dataset.

Subsequently, the frequency locations of all spectral peaks in l^{th} frame are identified to construct the following set $\mathbf{H}_l$.

$$\mathbf{H}_l = \{k : (|\mathbf{X}_l[k-1]| < |\mathbf{X}_l[k]|) \wedge (|\mathbf{X}_l[k]| > |\mathbf{X}_l[k+1]|)\} \tag{3.2}$$

Here, $0 \leq k < (N_f - 1)$ and $|\mathbf{X}_l[k]|$ indicates the magnitude of $\mathbf{X}_l[k]$. The number of spectral peaks in each frame varies. Not all spectral peaks are important for the SMC task. Thus, only a fixed number (at most p, say) of the highest amplitude peaks are selected from the spectrum of each frame. These highest spectral peaks from each frame are used to construct the truncated frequency location set $\tilde{\mathbf{H}}_l$. It is defined as

$$\tilde{\mathbf{H}}_l = \{k_{(1)}, k_{(2)}, \ldots k_{(q)}\} \qquad \left(\tilde{\mathbf{H}}_l \subseteq \mathbf{H}_l\right) \tag{3.3}$$

such that $|\mathbf{X}_l[k_{(0)}]| \geq |\mathbf{X}_l[k_{(1)}]| \geq \ldots \geq |\mathbf{X}_l[k_{(q)}]|$ and $0 < q \leq (p-1)$. If for any l^{th} frame, $q < (p-1)$, then the highest frequency location (i.e., $max(\tilde{\mathbf{H}}_l)$) in $\tilde{\mathbf{H}}_l$ is repeated $p - 1 - q$ times to maintain uniform cardinality of $\tilde{\mathbf{H}}_l$ (i.e., $\mid \tilde{\mathbf{H}}_l \mid = p$) for all frames. When just a small number of highest amplitude spectral peaks are considered ($p = 10$, say), this repeating procedure has a negligible effect on the peak amplitude and location distributions. It is evident from Table 3.2 that the percentage of frames requiring peak repetition is *Nil* for two of the datasets used for evaluation, (*GTZAN* and

Table 3.2: Repeating procedure statistics for $p = 10$.

	Percentage of peak-repeated frames		
Dataset	Music	Speech	Overall
GTZAN	0%	0%	0%
Scheirer-slaney	0%	0%	0%
Musan	0.022%	0.020%	0.021%

Scheirer-Slaney, see section 3.4), while only a minuscule percentage of frames from the third dataset (*MUSAN* dataset, section 3.4) require peak repetition.

The elements of $\tilde{\mathbf{H}}_l$ (frequency locations of the p highest peaks in the l^{th} frame spectra) are further sorted in descending order to construct the vector $\mathbf{fH}_l$ such that

$$\mathbf{fH}_l[0] \geq \mathbf{fH}_l[1] \geq \ldots \geq \mathbf{fH}_l[p-1] \tag{3.4}$$

Here, $\mathbf{fH}_l[r] \in \tilde{\mathbf{H}}_l$ and $r = 0, \ldots (p-1)$. $\mathbf{fH}_l$ contains the sorted frequency locations of the spectral peaks in $\tilde{\mathbf{H}}_l$. The vectors $\mathbf{fH}_l$ $(l = 0, \ldots (L-1))$ are used to construct a $p \times L$ Peak Location Matrix (PLM) $\mathcal{L}$ for an audio interval. The l^{th} column of $\mathcal{L}$ is defined as

$$\mathcal{L}_l = \mathbf{fH}_l^{\mathrm{T}} \tag{3.5}$$

Similarly, a Peak Amplitude Matrix (PAM) $\mathcal{A}$ can be constructed. The elements of $\mathcal{A}$ are defined as

$$\mathcal{A}[r, l] = \mathbf{X}_l[h] \tag{3.6}$$

where $h = \mathcal{L}[r, l]$, $r = 0, \ldots (p-1)$ and $l = 0, \ldots (L-1)$. A flowchart describing the procedure of computing the PLM (or PAM) matrix is provided in Fig. 3.4. Each row of $\mathcal{L}$ is defined as Location Sequence of Peak Traces (LSPT). Similarly, each row of $\mathcal{A}$ is defined as Amplitude Sequence of Peak Traces (ASPT). It may be noted that the first row of $\mathcal{L}$ (or $\mathcal{A}$) corresponds to the peak traces of the highest end of the spectrum. Similarly, the last row of $\mathcal{L}$ (or $\mathcal{A}$) corresponds to the peak traces of the lowest end of the spectrum. The LSPT and ASPT are sequences of peak location and amplitude values, respectively. These peak traces are viewed in this work as sub-channels of information extracted from the spectrogram of an audio interval.

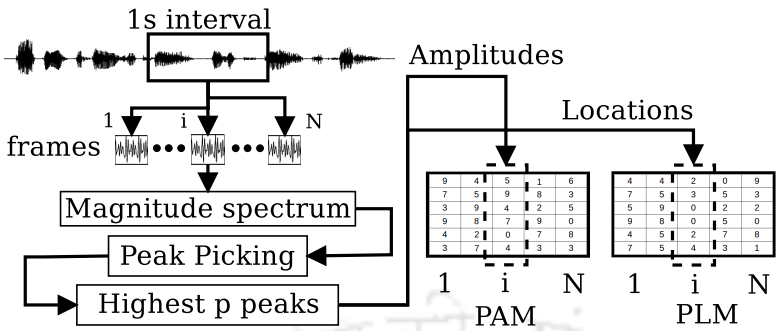


Figure 3.4: Flow chart illustrating the process of computing the PAM and PLM matrices.

In Fig. 3.2, the traces of identified spectral peaks are shown in the actual time-frequency scale as a separate representation and termed as SPT-spectrogram. An SPT-spectrogram is a matrix of the same shape as the actual spectrogram and initialized with zeros. Each spectral peak is located with its frequency bin and frame index in this matrix and initialized with its amplitude. When plotted as an image, this SPT-spectrogram shows the peak traces extracted from the corresponding spectrogram. It may be observed (Fig. 3.2) that the peak traces capture unique striation patterns of speech and music spectrograms. Each LSPT (or ASPT) represents a part of this striation information. Audio signals composed of many sound sources consist of multiple fundamental frequencies. Spectrograms of such signals are noisy due to the disturbance of harmonic patterns. However, the signal retains a basic property of its audio class. As can be observed in Fig. 3.2, both monophonic (Fig. 3.2(a)) and polyphonic (Fig. 3.2(b)) music contain relatively linear striation patterns. Whereas, single-speaker (Fig. 3.2(c)) and multi-speaker (Fig. 3.2(d)) speech have curvy striations. Since the basic assumption of this chapter is preserved even in the case of multiple F_0 signals, the proposed approach is still able to capture the required discriminative information for classification. Efficacy of the proposed SPT approach can be confirmed by observing the SPT spectrograms shown in Fig. 3.2(a)-(d) that contain all the prominent spectral striations for all four cases. It is worth noting that these SPT-spectrograms are generated just for visualization purposes and are not used for feature computation. The proposed features are computed using the PLM (and PAM) matrix. The proposal for modeling the distributions of LSPTs (or ASPTs) as features are discussed next.

3.3.2 Statistical moments of peak traces as feature

Fig. 3.3(a)-(c) respectively show distributions of first, fifth, and tenth LSPT distributions of speech and music across all data in the *GTZAN* dataset. Similarly, Fig. 3.3(d)-(f) respectively show the

first, fifth, and tenth ASPT distributions of speech and music. The LSPTs (and ASPTs) have been computed using short-term frames of size 10 milliseconds (ms) and a frameshift of 5 ms. It can be seen that the corresponding distributions of LSPT and ASPT of speech and music are different. This difference might be useful in classifying these two classes if the distributions of these sequences are represented in a suitable feature space. The first proposal involves using Mean and Standard Deviation (MSD) to model these distributions. Accordingly, the features extracted from PLM ($\mathcal{L}$, (equation 3.5)) and PAM ($\mathcal{A}$ (equation 3.6)) are named as MSD-LSPT and MSD-ASPT respectively. For notational convenience, the index r $(0 \leq r < (p-1))$ is used for referring to the r^{th} row of $\mathcal{L}$ and $\mathcal{A}$. Attributes derived from the r^{th} LSPT (or ASPT) will also be indexed by r. Mean $\mu_r^{\mathcal{L}}$ and standard deviation $\sigma_r^{\mathcal{L}}$ of the r^{th} LSPT is computed as follows.

$$\mu_r^{\mathcal{L}} = \frac{1}{L}\sum_{l=0}^{(L-1)} \mathcal{L}[r][l] \qquad \sigma_r^{\mathcal{L}} = \sqrt{\frac{1}{L}\sum_{l=0}^{(L-1)} (\mathcal{L}[r][l] - \mu_r^{\mathcal{L}})^2} \tag{3.7}$$

The MSD feature computed from PLM is proposed as a $2p$-dimensional vector given by

$$\text{MSD-LSPT} = \left[\mu_0^{\mathcal{L}}, \ldots \mu_{(p-1)}^{\mathcal{L}}, \sigma_0^{\mathcal{L}}, \ldots \sigma_{(p-1)}^{\mathcal{L}}\right] \tag{3.8}$$

Similarly, the mean ($\mu_r^{\mathcal{A}}$) and standard deviation ($\sigma_r^{\mathcal{A}}$) of ASPT can be computed from PAM and can be used to construct the MSD feature as a $2p$-dimensional vector given by

$$\text{MSD-ASPT} = \left[\mu_0^{\mathcal{A}}, \ldots \mu_{(p-1)}^{\mathcal{A}}, \sigma_0^{\mathcal{A}}, \ldots \sigma_{(p-1)}^{\mathcal{A}}\right] \tag{3.9}$$

A $4p$-dimensional feature vector MSD-ASPT-LSPT can be formed by concatenating MSD-ASPT and MSD-LSPT. The MSD features extracted from individual speech and music audio intervals are used for training classifiers. Performance of MSD features in SMC on standard datasets are shown in Section 3.4. This proposal uses only the first and second-order statistics of LSPT (or ASPT). The following proposal employs Gaussian mixture models for modeling peak location and amplitude distributions.

3.3.3 Component Bag-of-Words (CBoW) features from peak traces

Peak traces are temporally ordered sequences of prominent peaks occurring in successive frames of an audio interval. It is believed that these sequences can capture the highest energy striation patterns

observed in the spectrograms. It can be observed from Fig. 3.3 that the peak traces exhibit multi-modal distributions. Thus, using only mean and standard deviation might be insufficient to model these distributions. Moreover, the MSD features are extracted from individual audio intervals and are oblivious to their global distribution. This reasoning motivated the proposal of another set of features capable of representing the inherent multi-modality of the global distribution.

Gaussian mixture models (GMMs) are widely used to characterize multi-modal data. A K-component GMM $\mathcal{G} = \{\mathcal{C}_0, \mathcal{C}_1, \ldots \mathcal{C}_{(K-1)}\}$ consists of the component Gaussians $\mathcal{C}_j = \{\pi_j, \mu_j, \nu_j\}$ $(j = 0, \ldots (K-1))$. Here, π_j is the mixing parameter, μ_j is the mean, and ν_j is the variance of $\mathcal{C}_j$. Generally, a GMM is learned by using the Expectation-Maximization algorithm. The number of GMM components (K) is selected empirically based on experimental results. In this chapter, single-dimensional GMMs are trained with an optimal number of modes (K) to model the distribution of any r^{th} peak trace across the whole training dataset. Let $\mathcal{G}^r$ be a GMM trained on any r^{th} peak trace of either speech or music. Let u be the location (or amplitude) of a member peak of the r^{th} peak trace. The posterior probability of the j^{th} component $\mathcal{C}_j^r$ of $\mathcal{G}^r$ with respect to u can be computed as

$$\mathcal{P}\left(\mathcal{C}_j^r \mid u\right) = \frac{P\left(u \mid \mathcal{C}_j^r\right)\pi_j^r}{\sum_{i=0}^{(K-1)} P\left(u \mid \mathcal{C}_i^r\right)\pi_i^r} \tag{3.10}$$

Here, the likelihood function $P\left(u \mid \mathcal{C}_j^r\right)$ is defined as follows.

$$P\left(u \mid \mathcal{C}_j^r\right) = \frac{1}{\sqrt{2\pi\nu_j^r}}\, exp\left(-\frac{\left(u-\mu_j^r\right)^2}{2\nu_j^r}\right) \tag{3.11}$$

Let ${}_m\mathcal{G}^r = \left\{{}_m\mathcal{C}_j^r\right\}$ and ${}_s\mathcal{G}^r = \left\{{}_s\mathcal{C}_j^r\right\}$ $(j = 0, \ldots (K-1))$ be two GMMs learned from the r^{th} peak traces across the whole training set of music and speech respectively. The peak trace distributions of music and speech are observed to be distinctly different. This will lead to two GMMs with different component Gaussians. Thus, the following (given by equation 3.12) may be assumed for most cases.

$$\frac{1}{L}\sum_{l=0}^{(L-1)} \mathcal{P}\left({}_m\mathcal{C}_j^r \mid u_l\right) \neq \frac{1}{L}\sum_{l=0}^{(L-1)} \mathcal{P}\left({}_s\mathcal{C}_j^r \mid u_l\right) \tag{3.12}$$

Here, u_l are the location (or amplitude) of peak traces of an interval of L frames. Thus, the Component Bag-of-Words (CBoW) features are proposed as averaged K posterior probabilities obtained from speech and music GMMs learned from p peak traces. These features have been named such because of

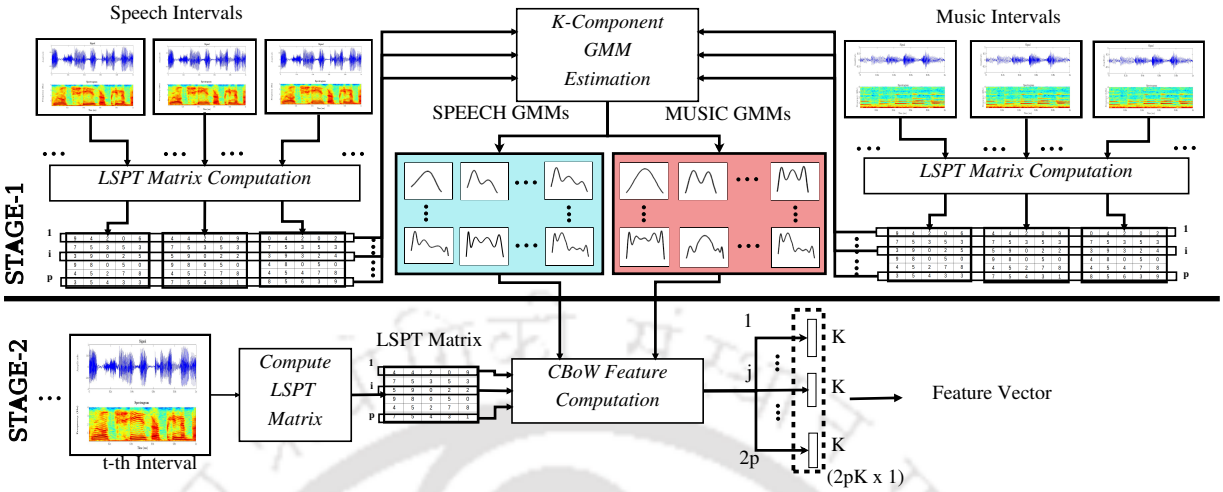


Figure 3.5: Schematic diagram representing the procedure for computing the CBoW-LSPT feature. The feature computation is a two-stage process. **STAGE 1** estimates separate GMMs for speech and music peak traces from entire training data. These learned GMMs are used in the **STAGE 2** to construct the CBoW-LSPT features. While computing CBoW-ASPT feature, the *LSPT Matrix Computation* block is replaced by the *ASPT Matrix Computation* block.

their similarity to bag-of-words representation existing in the literature. The CBoW feature extraction is a two-stage process. The first stage involves the estimation of separate GMMs from peak traces of all speech and music training data. This first stage is described next.

Let ${}_s\mathcal{L}^t$ be the PLM matrix constructed from the t^{th} interval ($t = 0, \ldots (T_s - 1)$) of speech training data. Similarly, let ${}_m\mathcal{L}^{\tilde{t}}$ denote the PLM matrix constructed from the $\tilde{t}^{th}$ interval ($\tilde{t} = 0, \ldots (T_m - 1)$) of music training data. Let the r^{th} rows of ${}_s\mathcal{L}^t$ and ${}_m\mathcal{L}^{\tilde{t}}$ be denoted by ${}_{ls}R_t^r$ and ${}_{lm}R_{\tilde{t}}^r$ respectively ($r = 0, \ldots (p-1)$). The sets ${}_{ls}\mathbf{S}^r$ and ${}_{lm}\mathbf{S}^r$ are constructed (equations 3.13 and 3.14) for accumulating the frequency locations of the r^{th} peak traces of respective speech and music training data.

$$ {}_{ls}\mathbf{S}^r = \{{}_{ls}R_t^r[i]; t = 0, \ldots (T_s - 1), i = 0, \ldots (L-1)\} \tag{3.13} $$

$$ {}_{lm}\mathbf{S}^r = \left\{{}_{lm}R_{\tilde{t}}^r[i]; \tilde{t} = 0, \ldots (T_m - 1), i = 0, \ldots (L-1)\right\} \tag{3.14} $$

Single dimensional K-component GMMs ${}_{ls}\mathcal{G}^r$ and ${}_{lm}\mathcal{G}^r$ are estimated from the elements of ${}_{ls}\mathbf{S}^r$ and ${}_{lm}\mathbf{S}^r$ respectively. It may be noted that two GMMs are learned for any r^{th} peak trace. Thus, $2p$ GMMs are estimated for p peak traces. The learned GMMs are used to compute posterior probability vectors for any given audio interval. The extraction of CBoW features as posterior probability vectors are described next.

Let $\mathcal{L}$ be the $p \times L$ PLM matrix of an audio interval containing L frames. Let, ${}_{l}R^{r}$ be the r^{th} row of $\mathcal{L}$. The learned GMMs ${}_{lm}\mathcal{G}^{r}$ and ${}_{ls}\mathcal{G}^{r}$ are used to obtain component-wise posterior probabilities for each element of ${}_{l}R^{r}$. The averaged posterior probability vector ${}_{lm}\mathcal{H}^{r}$ for ${}_{l}R^{r}$ is obtained by using ${}_{lm}\mathcal{G}^{r}$. This is computed using equations 3.15 and 3.16.

$$ {}_{lm}\mathcal{Z}^{r}\left(i\right) = \left[\mathcal{P}\left({}_{lm}\mathcal{C}_{0}^{r} \mid {}_{l}R^{r}[i]\right), \ldots \mathcal{P}\left({}_{lm}\mathcal{C}_{(K-1)}^{r} \mid {}_{l}R^{r}[i]\right)\right] \tag{3.15} $$

$$ {}_{lm}\mathcal{H}^{r} = \frac{1}{L}\sum_{i=0}^{(L-1)} {}_{lm}\mathcal{Z}^{r}\left(i\right) \tag{3.16} $$

Similarly, the averaged posterior probability vector ${}_{ls}\mathcal{H}^{r}$ for ${}_{l}R^{r}$ is computed (using ${}_{ls}\mathcal{G}^{r}$) according to equations 3.17 and 3.18.

$$ {}_{ls}\mathcal{Z}^{r}\left(i\right) = \left[\mathcal{P}\left({}_{ls}\mathcal{C}_{0}^{r} \mid {}_{l}R^{r}[i]\right), \ldots \mathcal{P}\left({}_{ls}\mathcal{C}_{(K-1)}^{r} \mid {}_{l}R^{r}[i]\right)\right] \tag{3.17} $$

$$ {}_{ls}\mathcal{H}^{r} = \frac{1}{L}\sum_{i=0}^{(L-1)} {}_{ls}\mathcal{Z}^{r}\left(i\right) \tag{3.18} $$

It is worth noting that, both ${}_{ls}\mathcal{H}^{r}$ and ${}_{lm}\mathcal{H}^{r}$ are K length vectors. The proposed CBoW-LSPT feature is constructed as a $2 \times K \times p$ dimensional vector and is given by

$$ \text{CBoW-LSPT} = \left[\, {}_{lm}\mathcal{H}^{0},\ {}_{ls}\mathcal{H}^{0}, \ldots,\ {}_{lm}\mathcal{H}^{(p-1)},\ {}_{ls}\mathcal{H}^{(p-1)}\right] \tag{3.19} $$

Similarly, PAM matrices computed from both speech and music intervals are denoted as ${}_{m}\mathcal{A}^{\tilde{t}}$ and ${}_{s}\mathcal{A}^{t}$ respectively. The r^{th} rows ${}_{am}R_{\tilde{t}}^{r}$ and ${}_{as}R_{t}^{r}$ of ${}_{m}\mathcal{A}^{\tilde{t}}$ and ${}_{s}\mathcal{A}^{t}$ are used to form the sets ${}_{am}\mathbf{S}^{r}$ and ${}_{as}\mathbf{S}^{r}$. The respective GMMs ${}_{am}\mathcal{G}^{r}$ and ${}_{as}\mathcal{G}^{r}$ are estimated from ${}_{am}\mathbf{S}^{r}$ and ${}_{as}\mathbf{S}^{r}$. For any given audio interval with PAM $\mathcal{A}$, the averaged posterior probability vectors ${}_{am}\mathcal{H}^{r}$ and ${}_{as}\mathcal{H}^{r}$ are computed in a similar manner as described in equations 3.16 and 3.18. The CBoW-ASPT feature is constructed as a $2 \times K \times p$ length vector and is given by

$$ \text{CBoW-ASPT} = \left[\, {}_{am}\mathcal{H}^{0},\ {}_{as}\mathcal{H}^{0}, \ldots,\ {}_{am}\mathcal{H}^{(p-1)},\ {}_{as}\mathcal{H}^{(p-1)}\right] \tag{3.20} $$

Finally, a $4 \times K \times p$-dimensional feature vector CBoW-ASPT-LSPT can be formed by concatenating CBoW-ASPT and CBoW-LSPT. Fig. 3.5 shows a functional block diagram for computing the CBoW features. Detailed experimentation with proposed features and the performance analysis results are

presented next.

3.4 Experiments and results

The proposed features are validated on six datasets. These are (a) *GTZAN* Music/Speech collection [204], (b) *Scheirer-Slaney* Music-Speech Corpus [99], (c) *MUSAN* - A Music, Speech and Noise corpus [203], (d) *Movie-MUSNOMIX* (section 3.2), (e) *DAFx-12* [206], and (f) *Muspeak* [205]. Both *GTZAN* and *Scheirer-Slaney* contain 1 hour of data. On the other hand, *MUSAN* is a much larger dataset containing around 102.5 hours of speech and music data. The *Movie-MUSNOMIX* dataset consists of approximately 3 hours 40 minutes of speech and music signals. The *DAFx-12* and *Muspeak* datasets consist of continuous audio sequences with natural transitions between speech and music. The last two datasets are used to evaluate the robustness of proposed features to real-world signals.

Five baseline approaches have been used to benchmark the proposal in this chapter. The first baseline uses speech specific feature set (FS) proposed by Khonglah et al. [148] (Khonglah-FS). The second baseline (Sell-FS) uses chroma-based features to represent music tonality for enhanced speech music classification [140]. The MFCCs are widely used in most speech processing applications, including SMC [149]. This chapter uses the 13-dimensional MFCCs along with their 13 Δ (velocity) and 13 $\Delta\Delta$ (acceleration) coefficients as the third baseline (MFCC-39) for performance comparison. Keum et al. in [114] proposed features derived from a variant of spectral peak tracking for speech and music discrimination. This approach is adopted as the fourth baseline (Keum-FS). Finally, this chapter uses the CNN architecture proposed by Papakostas et al. [212] as the fifth baseline (Papakostas-CNN) to benchmark the proposal against contemporary deep network based methods. Spectrogram images of 1 second intervals are used to train the CNN and generate results to compare with the proposed approach.

The experiments are performed using standard python packages [2]. This chapter uses a train-test split ratio of 80 : 20. The examples in each of the two sets are sampled randomly without replacement to ensure no overlap between the two sets. Classifier hyperparameters have been tuned over a validation set extracted from the training set, keeping the testing set untouched until final evaluation. Classification performance is reported using the mean and standard deviation of F1-scores [230] obtained from 10 independent trials. Based on experimental results, the number of peak traces (p) and the number of GMM mixtures (K) for computing the proposed features are set to 10

[2]Codes used in this chapter can be found at https://github.com/mrinmoy-iitg/Speech-Music-Classification-Using-SPT

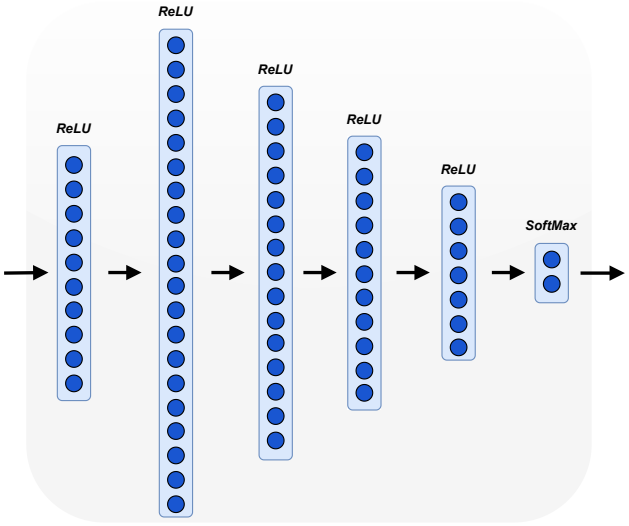


Figure 3.6: Architechture of DNN used in Table 3.6, 3.8 and 3.9.

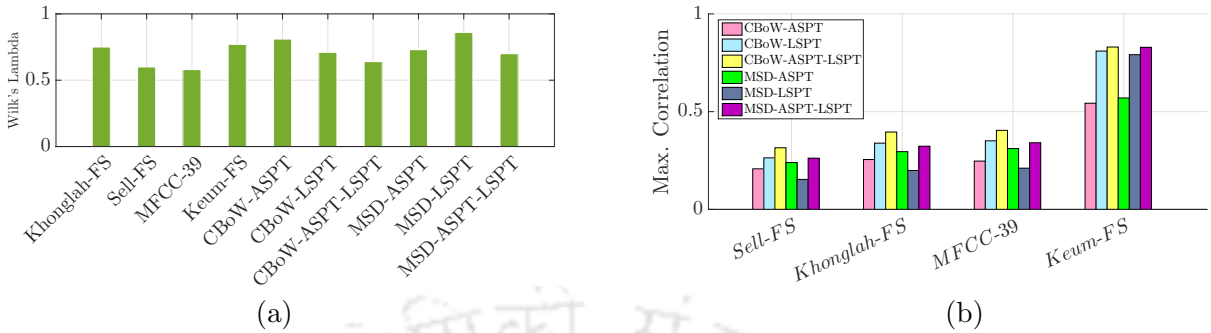
and 5, respectively. For *GTZAN* and *Scheirer-Slaney* datasets, classification results for baseline, and proposed features are generated using SVM with a Radial Basis Function (RBF) kernel. The cost and RBF kernel bandwidth (γ_{rbf}) parameters of SVM are tuned using a grid search. For *MUSAN* dataset, the results for baseline and proposed features are computed using bagged RBF-SVM and Deep Neural Network (DNN) based classifier. The bagged SVM classifier ensemble has 10 base SVM classifiers with 20% bootstrap in each bag. For the *Movie-MUSNOMIX*, *DAFx-12* and *Muspeak* datasets, the DNN classifier is used to generate results. The cost and γ_{rbf} parameters of all base SVMs are optimized using a grid search. Fig. 3.6 shows the DNN architecture used in this chapter. The DNN model is trained for 100 epochs with a batch size of 64. The network is trained with the Adam [231] optimizer with an initial learning rate of 10^{-4}.

3.4.1 Statistical significance test

The statistical significance of features is studied in this chapter using Multivariate ANalysis Of VAriance (MANOVA) [232]. The Wilks' Lambda (Λ) is a test statistic used in MANOVA. The value of Λ is used to test a null hypothesis (H_0) against an alternative hypothesis (H_1). These hypotheses are defined below.

- The null hypothesis H_0 states that different data classes belong to a same cluster having a common mean for a set of dependent variables.

- The alternative hypothesis H_1 refutes H_0 by claiming that each of the classes in the data corre-



(a) (b)

Figure 3.7: Subfigure (a) illustrates the statistical significance of the baseline and the proposed features in terms of Wilks' Lambda (Λ) values obtained by performing MANOVA over the MUSAN dataset. A feature set with a lower value of Λ is preferred. Subfigure (b) illustrates the maximum CCA values between every pair of baseline and proposed feature sets for the MUSAN dataset. The proposed feature sets are statistically significant and carry complementary information to the baseline feature sets.

sponds to different clusters (and different means) for the dependent variables.

The Λ values of different feature sets illustrated in Fig. 3.7(a) indicate that the proposed features have comparable statistical significance with the baseline features in the current task. Moreover, the combination of amplitude and location information in the CBoW-ASPT-LSPT and MSD-ASPT-LSPT improve the statistical significance of the feature set. Therefore, the proposed features can be useful for the current SMC task.

This chapter also uses Canonical Correlation Analysis (CCA) to gauge the feasibility of combining different baseline and proposed feature sets. The low correlation of proposed features with Sell-FS, Khonglah-FS, and MFCC-39, as observed from the CCA values (Fig. 3.7(b)), indicate that these feature sets have considerable complementary information. It may also be noted that due to the similar information captured by the proposed features and Keum-FS, these feature sets have a high correlation. Nevertheless, the complementary information between the proposed features and three baseline feature sets (Sell-FS, Khonglah-FS, and MFCC-39) suggests they might be good candidates for feature fusion to improve performance. In the subsequent subsections, the results of experiments performed with the proposed features are discussed.

3.4.2 Effect of varying frame and interval size

Table 3.3 presents the effect of changing short-term audio window size from 10 ms to 30 ms (frame sizes are taken as half of the window sizes). A shorter frame gives a smoother spectrum that resembles the formant structure in speech. The presence of formants in speech discriminates it from music.

Table 3.3: Performance of proposed features for *different audio frame sizes* over *GTZAN* dataset using 10-component GMM classifier. The interval size is fixed at 1s. Frame sizes are taken as 5 ms, 10 ms, and 15 ms, corresponding to window sizes of 10 ms, 20 ms, and 30 ms, respectively.

Features	Frame Size (in miliseconds)		
	10	20	30
MSD-ASPT	86.96 ±1.65	86.42 ±1.38	86.73 ±1.91
MSD-LSPT	87.51 ±0.89	88.68 ±0.85	86.60 ±1.39
MSD-ASPT-LSPT	90.92 ±1.24	91.21 ±0.92	89.41 ±1.38
CBoW-ASPT	86.56 ±1.28	86.98 ±1.74	88.53 ±1.29
CBoW-LSPT	87.51 ±2.16	87.40 ±1.35	85.79 ±2.16
CBoW-ASPT-LSPT	92.67 ±0.84	93.10 ±0.93	91.79 ±1.06

Table 3.4: Performance of proposed features for *different interval sizes* over *GTZAN* dataset using 10-component GMM classifier.

Features	Classification Interval Size (in seconds)		
	0.50	1.00	2.00
MSD-ASPT	84.34 ±1.37	86.96 ±1.65	86.83 ±2.08
MSD-LSPT	82.77 ±1.52	87.51 ±0.89	90.70 ±1.28
MSD-ASPT-LSPT	88.19 ±1.55	90.92 ±1.24	92.81 ±1.90
CBoW-ASPT	83.48 ±1.03	86.56 ±1.28	89.55 ±2.11
CBoW-LSPT	82.63 ±1.61	87.51 ±2.16	90.57 ±1.37
CBoW-ASPT-LSPT	89.36 ±1.01	92.67 ±0.84	94.16 ±1.68

Thus, the performance of the proposed features is expected to be better for smaller window sizes. The performance of the proposed features drops for a window size of 30 ms. On the other hand, almost similar performances are noted for window sizes of 10 ms and 20 ms. Hence, this chapter uses a 10 ms window size with a frame size of 5 ms.

Table 3.4 presents the performance of the proposed features computed for three different audio interval sizes – 0.5 s, 1 s and 2 s. An improvement in classification performance can be observed for an increase in interval size. This result indicates that using larger interval sizes leads to better modeling of the spectral peak traces. However, interval sizes of 2 s or more might smooth out instances of sharp transitions between audio categories in continuous audio streams. At the same time, a 0.5 s context would provide poor performance. Hence, this chapter considers 1 s audio intervals as units for classification decisions as a compromise.

Table 3.5: Performance of SMC using **SVM** (RBF kernel) classifier on *GTZAN* and *Scheirer-Slaney* datasets. The performances are reported in terms of the mean and standard deviation of the $F1$-score. The top three performances are indicated by: ★ (Best), ♡ (2^{nd} Best) and ♣ (3^{rd} Best).

Dataset	Baseline				Proposed					
	Khonglah-FS	Sell-FS	MFCC-39	Keum-FS	MSD-ASPT	MSD-LSPT	MSD-ASPT-LSPT	CBoW-ASPT	CBoW-LSPT	CBoW-ASPT-LSPT
GTZAN	91.02 ±1.53	**94.00 ±0.86** ♣	93.48 ±0.94	90.75 ±1.34	92.00 ±2.35	89.05 ±2.47	**94.17 ±2.19** ♡	92.58 ±2.24	92.46 ±2.90	**95.25 ±2.24** ★
Scheirer Slaney	**95.22 ±0.86** ♣	94.99 ±0.63	93.86 ±1.20	88.35 ±1.54	94.33 ±1.87	92.75 ±0.85	**96.12 ±1.78** ♡	95.03 ±1.15	93.89 ±1.40	**96.15 ±1.09** ★

Table 3.6: Performance of SMC using **Bagged-SVM** (RBF kernel) and **DNN** classifiers on *MUSAN* dataset. Performance is reported as: Average F1-score ± standard deviation. The top three performances are indicated by: ★ (Best), ♡ (2^{nd} Best) and ♣ (3^{rd} Best).

Classifier	Baseline				Proposed					
	Khonglah-FS	Sell-FS	MFCC-39	Keum-FS	MSD-ASPT	MSD-LSPT	MSD-ASPT-LSPT	CBoW-ASPT	CBoW-LSPT	CBoW-ASPT-LSPT
Bagged-SVM	91.09 ±0.12	97.32 ±0.05	**98.29 ±0.05** ♡	95.37 ±0.06	94.28 ±0.08	93.25 ±0.07	**98.10 ±0.05** ♣	95.32 ±0.08	96.75 ±0.09	**98.99 ±0.04** ★
DNN	92.49 ±0.12	97.62 ±0.09	**98.56 ±0.09** ♡	95.40 ±0.52	94.52 ±0.59	91.88 ±1.56	**97.51 ±0.55** ♣	95.35 ±0.67	97.14 ±0.63	**98.87 ±0.25** ★

Table 3.7: Performance of SMC using **DNN** classifier on *Movie-MUSNOMIX* dataset. Performance is reported as: Average F1-score ± standard deviation. The top three performances are indicated by: ★ (Best), ♡ (2^{nd} Best) and ♣ (3^{rd} Best).

Baseline				Proposed						
Khonglah-FS	Sell-FS	MFCC-39	Keum-FS	MSD-ASPT	MSD-LSPT	MSD-ASPT-LSPT	CBoW-ASPT	CBoW-LSPT	CBoW-ASPT-LSPT (EF)	CBoW-ASPT-LSPT (LF)
96.09 ±1.35 ♡	95.42 ±1.21	95.57 ±3.85	87.96 ±3.41	42.86 ±14.37	56.06 ±19.45	93.38 ±2.05	94.31 ±2.42	92.12 ±3.47	**95.78 ±2.61** ♣	**96.25 ±1.95** ★

3.4.3 Performance analysis

Table 3.5 presents performance of baseline and proposed features using SVM (RBF kernel) over the *GTZAN* and *Scheirer-Slaney* datasets. The CBoW-ASPT-LSPT and MSD-ASPT-LSPT features

Table 3.8: Peformance comparison of best baseline and proposed features with 2**D CNN** based baseline (Papakostas-CNN).

Dataset	Papakostas-CNN	Best baseline	Best Proposed (CBoW-ASPT-LSPT)
GTZAN	89.76 ±3.16	94.00 ±0.86 (Sell-FS)	95.25 ±2.24
Scheirer-Slaney	90.85 ±4.29	95.22 ±0.86 (Khonglah-FS)	96.15 ±1.09
Musan	99.36 ±0.76	98.56 ±0.09 (MFCC-39)	98.87 ±0.25

provide the best and second-best performance over these two datasets. Table 3.6 shows the classification performance of baseline and proposed features on *MUSAN* dataset using bagged SVM (RBF kernel) and DNN. All baseline and proposed features show better performance over this dataset due to the availability of a large amount of training data. Even the standard deviations of $F1$-scores are observed to reduce significantly. The MFCC-39 turns out to be the best baseline feature. The CBoW features individually perform better than individual MSD features. The MSD-ASPT-LSPT feature substantially improves upon the MSD features taken separately. However, CBoW-ASPT-LSPT stands out as the overall best performer with the DNN classifier. Table 3.7 illustrates the performance of the proposed features on the *Movie-MUSNOMIX* dataset. The proposed features also perform better than the baselines on the movie audio signals, which are known to have additional complexities. The early-fusion (EF) strategy of combining CBoW-ASPT and CBoW-LSPT (CBoW-ASPT-LSPT (EF) in Table 3.7) performs slightly poorer than the best baseline, Khonglah-FS. However, in a late-fusion (LF) strategy (CBoW-ASPT-LSPT (LF) in Table 3.7), the proposed features provide the best performance on the *Movie-MUSNOMIX* dataset as well. The MSD-ASPT and MSD-LSPT features are considerably poor performers. However, their early-fusion combination MSD-ASPT-LSPT performs well.

Table 3.8 presents a comparative performance analysis of Papakostas-CNN, the best one among baselines and the best one among proposed ones. This analysis is presented for the individual datasets. For *GTZAN* and *Scheirer-Slaney*, the performance of Papakostas-CNN is significantly lower than the best baselines (Sell-FS and Khonglah-FS, respectively) and proposed feature (CBoW-ASPT-LSPT). This decreased performance can be attributed to insufficient training data available in smaller datasets. However, for the larger *MUSAN* dataset, Papakostas-CNN outperforms all other methods. The proposed CBoW-ASPT-LSPT feature provides comparable performance on the *MUSAN* dataset. Such

Table 3.9: Result of combining the proposed features with contemporary deep network based techniques.

Feature	F1-score
CBoW-ASPT-LSPT (CAL)	98.87 ±0.25
DBF [233]	98.87 ±0.41
Papakostas-CNN-Embed [212]	99.17 ±0.2
CAL + DBF (Early fusion)	99.50 ±0.17
CAL + DBF (Late fusion)	99.61 ±0.15
CAL + Papakostas-CNN-Embed (Early Fusion)	99.66 ±0.06
CAL + Papakostas-CNN-Embed (Late Fusion)	99.80 ±0.05

a result indicates the efficiency of proposed CBoW features in SMC.

Experiments were also performed to show the effectiveness of combining the CBoW-ASPT-LSPT feature with two contemporary deep network based methods. First is the deep bottleneck feature (DBF), which has gained popularity in many speech processing applications in recent times [233]. The DBFs are generated using a deep neural network. One of the hidden layers in this network, the bottleneck layer, has significantly fewer nodes than other layers. Embeddings generated from this layer are referred to as DBF. The DBF network considered in this chapter has 5 hidden layers, and the middle one is the bottleneck layer. The bottleneck layer has a size of 50, the input and other hidden layers have 1313 nodes each, and the output layer has 2 nodes. MFCC (13-dimensional) features for every frame in a 1 second interval are concatenated and passed as input to the DBF network. Second, feature embeddings generated from the Papakostas-CNN network (Papakostas-CNN-Embed) are used as the other deep-learning based feature. Papakostas-CNN-Embed feature is extracted from the penultimate layer of the CNN network proposed in [212] and has a dimension of 4096. The DNN architecture illustrated in Fig. 3.6 is used as the classifier in this experiment.

The results obtained from these experiments are listed in Table 3.9. It can be observed that Papakostas-CNN-Embed performs better than DBF and CBoW-ASPT-LSPT. However, the results improve when the DBF and Papakostas-CNN-Embed features are separately combined with CBoW-ASPT-LSPT. Thus, the proposed CBoW-ASPT-LSPT features capture complementary information, which leads to improvement in performance upon combination. Both early and late fusion strategies improve classification performance. However, it is observed that late fusion provides better results in both cases.

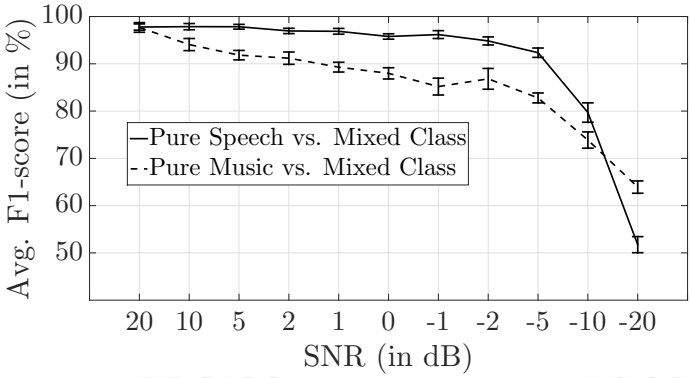


Figure 3.8: Figure illustrating the performance of CBoW-ASPT-LSPT with mixed class (MC) data. The performance drops drastically for both the cases when SNR increases beyond 2 dB

The efficiency of CBoW features may be intuitively explained as follows. First, each one-dimensional element of LSPT (or ASPT) is projected to a K-dimensional posterior probability vector. Such a non-linear transformation to a higher dimensional space might induce the separability of features. Second, the averaged posterior probability vectors of all peak traces are concatenated together. This concatenation further enhances the chances of separability in a higher-dimensional space. Third, averaging posterior probability vectors over an interval possibly emphasize the importance of class-specific components.

Table 3.8 indicates that the proposed feature CBoW-ASPT-LSPT outperforms Papakostas-CNN on the two smaller datasets (*GTZAN* and *Scheirer-Slaney*). However, Papakostas-CNN marginally outperforms CBoW-ASPT-LSPT on a larger dataset (*MUSAN*) by approximately 0.5%. This performance gain of Papakostas-CNN may be attributed to the availability of a larger amount of training data. Nevertheless, when embeddings obtained from Papakostas-CNN are combined with CBoW-ASPT-LSPT, the best performance on the *MUSAN* dataset is obtained (see Table 3.9). Thus, Papakostas-CNN-Embed and the CBoW-ASPT-LSPT feature may be a better combination for SMC on larger datasets. In contrast, the CBoW-ASPT-LSPT feature alone may be more suitable for the SMC task on smaller datasets.

3.4.4 Analysis of failure cases

In practical scenarios, classification models trained on clean speech and music data may be tested with data that have a mixture of both classes (SpMu). Accordingly, a set of experiments are performed to gauge the effectiveness of the proposed features in a scenario involving SpMu data. The trained

models are tested against clean speech and SpMu data in one set. The SpMu test data are synthetically generated by mixing speech with music at different Signal-to-Noise Ratios (SNR) (−20 dB to +20 dB). The mixing SNR is computed by considering music as the reference signal and speech as the mixing noise. This type of SpMu signal is termed Mixed Class with Speech Noise (MCSN), and the experiment is termed Clean Speech vs. MCSN (CS-MCSN).

Similarly, the trained models are also used to classify clean music and SpMu data in the second set of experiments. The SpMu test data for this experiment is synthetically generated by mixing music with speech signals at different SNR (−20 dB to +20 dB). This SpMu signal is termed Mixed Class with Music Noise (MCMN), where speech is considered the reference signal and music is mixed as noise. The second set of experiments is termed Clean Music vs. MCMN (CM-MCMN). The performance obtained for these experiments is illustrated in Fig. 3.8.

It can be observed from Fig. 3.8 that as the energy of mixing noise in the SpMu data increases for either case, the performance gradually drops. With the increasing energy of the mixing noise, SpMu data becomes increasingly similar to the clean class data considered in the experiment. For example, in the CS-MCSN case, MCSN data becomes increasingly similar to speech with increasing energy of the mixing noise (speech) from +20 dB to −20 dB. The same reasoning works for the CM-MCMN case. MCMN data becomes increasingly similar to music with increasing energy of the mixing noise (music) from +20 dB to −20 dB. Hence, most test MCMN samples are recognized as music in the CM-MCMN experiment at lower SNR. Similarly, the MCSN samples are predicted as speech in the CS-MCSN experiment at lower SNR. The lowest performance reaching close to 50% justifies this reasoning. Hence, it can be observed from Fig. 3.8 that the proposed feature shows stable performance with graceful degradation as long as either speech or music is the dominating content over the added noise in mixed class data. Thus, it can be said that the proposed feature is robust up to a tolerable noise level.

3.4.5 Performance with real-world signals

Table 3.10 lists the performances of the proposed CBoW features on the *DAFx-12* dataset [206] that contains real-world speech and music signals. The *DAFx-12* dataset consists of mixed speech and music signals which pose an additional challenge to the proposed approach. Moreover, there is a significant data imbalance between the speech and music classes. The amount of music present in the *DAFx-12* dataset is more than three times that of speech. Such complexities in the dataset are

Table 3.10: Performance of the proposed approach on real signals from the *DAFx12-dataset* [206] are tabulated here. Baseline results are quoted directly from the reference.

		Music/ Non-music				Speech/ Non-speech			
Test set	**Method**	**Acc**	**Prec**	**Rec**	**F1**	**Acc**	**Prec**	**Rec**	**F1**
Swiss	Schlüter et al. [206]	97.30	98.80	98.00	98.40	98.40	96.40	96.50	96.40
	CBoW-ASPT	91.37	97.13	92.54	94.78	93.24	80.79	92.80	86.38
	CBoW-LSPT	90.86	96.41	92.66	94.5	92.68	82.09	87.36	84.64
	CBoW-ASPT-LSPT	93.10	96.65	95.14	95.89	94.99	86.61	92.60	89.51
Austrian	Schlüter et al. [206]	95.60	95.30	97.40	97.30	97.00	95.90	95.10	95.50
	CBoW-ASPT	89.65	95.89	91.34	93.56	90.15	87.98	82.54	85.17
	CBoW-LSPT	88.71	94.76	91.33	93.01	89.39	90.32	77.34	83.33
	CBoW-ASPT-LSPT	91.17	94.85	94.38	94.61	90.15	92.95	77.11	84.29

reflected in the performance of proposed CBoW features tabulated in Table 3.10. The CBoW-ASPT-LSPT is again the best performer among the proposed features over both the test sets provided with the *DAFx-12* dataset. However, the baseline performance (Schlüter et al. [206]) is better than the proposed features. The proposed features provide comparable performance in the music detection task. Because of the extreme data imbalance in the dataset, the speech detection performance of the proposed features is quite poor. The proposed features, designed for isolated speech and music signals, are also affected by the presence of mixed speech and music signals in the *DAFx-12* dataset. Nevertheless, the proposed features show promising results with real-world signals.

Table 3.11 shows the performance of the proposed CBoW features on the *Muspeak* [205] dataset that is used for model training in the popular MIREX challenges. The submissions to MIREX are evaluated on undisclosed datasets. However, the *Muspeak* dataset consists of audio signals with natural transitions between speech and music. Hence, this dataset is considered in this chapter to evaluate the performance of the proposed features. The baseline results of Doukhan et al. [2] reported in Table 3.11 cannot be directly compared to that of the proposed features because of the difference in evaluation datasets. The baseline performances can be best-considered references to judge the efficacy of proposed features. The previously discussed *DAFx-12* dataset contains a similar type of audio data as those present in the *Muspeak* dataset. Hence, the models trained on the *DAFx-12* dataset have been used to evaluate the audio from the *Muspeak* dataset. The results are reported as $F1$-score

Table 3.11: Event-level performance on Muspeak dataset [205]. The onset-offset $F1$-score at different tolerance durations is reported.

		Dataset	**F1-score** (500 ms)	**F1-score** (1000 ms)
Music	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.0930	0.1142
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.2235	0.2480
	CBoW-ASPT	Muspeak [205]	0.0794	0.0922
	CBoW-LSPT		0.0864	0.1047
	CBoW-ASPT-LSPT		0.0868	0.0970
Speech	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.1603	0.2122
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.4139	0.4350
	CBoW-ASPT	Muspeak [205]	0.1821	0.1918
	CBoW-LSPT		0.1858	0.2078
	CBoW-ASPT-LSPT		0.1047	0.2124

of correct detection of onset or offset points of speech and music in the audio sequences within a specific tolerance window (500 ms or 1000 ms). The onset and offset points are termed events in the literature, and the $F1$-score of detecting such events is known as event-level performance. The reported performances indicate that the EF of CBoW-ASPT and CBoW-LSPT (CBoW-ASPT-LSPT) performs best among the proposed features. The event-level performance is observed to improve with a larger tolerance window. Satisfactory performance is obtained with the proposed features in detecting natural transitions of speech and music in audio signals.

3.5 Summary

This chapter proposes a novel two-stage feature extraction scheme for representing the time-frequency characteristics of an audio interval. The first stage detects a fixed number of prominent spectral peak traces in an audio interval. Two proposed features (MSD and CBoW) are computed from the detected peak traces' locations (or amplitudes). The performance of the proposed algorithm is validated on six datasets and compared with five baseline approaches. It is shown that the fusion of either MSD (i.e., MSD-ASPT-LSPT) or CBoW (i.e., CBoW-ASPT-LSPT) features provide better performance than the individual ones. Experiments show that CBoW-ASPT-LSPT stands out as the

overall best feature. Furthermore, combining the proposed CBoW-ASPT-LSPT feature with contemporary deep bottleneck features and deep CNN embeddings improves the classification performance, indicating that such a combination can form a highly robust SMC system. The proposed features also perform well in detecting speech and music in real-world audio signals.

This chapter employs only the magnitude spectrum to compute the proposed features. The phase spectrum of speech and music signals is not sufficiently explored in the literature. Hence, the subsequent chapter attempts to study the distinct phase characteristics of these signals. Moreover, the utility of the phase component in discriminating between speech and music signals is also established through extensive experimentation.

4

Phase Features for Speech Music Classification

Publications

- **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Speech/music classification using phase-based and magnitude-based features", in *Speech Communication*, vol. 142, pp. 34-48, 2022.

Contents

Objective

It has been discussed in the previous chapter that speech and music constitute a significant proportion of the movie audio. Speech vs. Music Classification (SMC) is an essential preprocessing step for many high-level audio-based movie content analysis applications, like audio diarization, subtitle generation, and dynamic emotion prediction. In that regard, this chapter proposes another approach for SMC. Similar to the previous chapter's proposal, researchers have mainly used various magnitude-based features for SMC in the past. Comparatively, the phase spectrum has received lesser attention in this task. The phase component is believed to carry valuable information to help determine its audio class. This chapter explores three existing phase-based features for SMC. The potential of phase information is highlighted through a statistical significance test and canonical correlation analysis. The proposed approach is benchmarked against four baseline magnitude-based feature sets. The contributed Movie-MUSNOMIX *dataset and widely used public datasets like* MUSAN, GTZAN, Scheirer-Slaney, DAFx-12 *and* Muspeak *have been used for performance evaluations. In combination with magnitude-based features, phase-based features improve the baseline performance consistently for the datasets used. Various phase and magnitude-based feature combinations also show satisfactory generalization over the datasets. Phase-based features perform satisfactorily in identifying speech and music signals corrupted with different environmental noises at various SNR levels. Moreover, the combination of phase-based and magnitude-based features is effective in segmenting continuous speech and music signal sequences.*

4.1 Task overview

Efficient detection of speech and music signals is an essential preprocessing step of high-level applications for movie content analysis like audio diarization, subtitle generation, and dynamic emotion prediction [234, 235]. Austin et al. [24] showed that detecting high-intensity music in movies could aid the prediction of their genre. Precise speech detection is vital for enhancing the intelligibility of movie dialogues when listening in noisy environments [236]. Therefore, it is necessary to develop systems that can efficiently detect speech and music intervals and generalize well to the challenging conditions of movie audio signals. This chapter proposes an SMC system that employs comparatively lesser explored phase information of the audio signal as a feature for classification.

Speech and music signals have distinct acoustic properties [127, 208]. A challenge in building efficient classifiers for these signals is identifying features that capture relevant discriminatory in-

formation. Various audio features have been explored in the literature to capture such differences. Standard spectral features like Spectral Centroid, Spectral Rolloff, and Spectral Flux [149] are widely used. Temporal features like Zero Crossing Rate (ZCR) [140], Energy [139], Entropy [137] and Root Mean Square values [140] are also popular. Speech-specific features have been shown to efficiently discriminate speech from music [148]. Sell et al. [140] have shown that music-specific chroma-based features with other standard features work very well in this task. Other approaches in Speech vs. Music Classification (SMC) literature extract features from the spectrogram representation [115, 139]. Recently, deep-network based automatic feature learning techniques have also been explored that provide higher success rates [211, 212].

The popular Music Information Retrieval Evaluation eXchange (MIREX) challenges have been organized every year since 2005 by the International Music Information Retrieval Systems Evaluation Laboratory at the University of Illinois at Urbana-Champaign, United States of America. In the 2015 and 2018 editions of the MIREX challenge, one of the tasks was speech and music detection. Many state-of-the-art submissions competed for the top positions in the task. A brief review of the best submissions for this task is provided here. The SMC systems submitted to these challenges can be grouped into two broad categories. Systems of the first category start by pre-segmenting the input audio into homogeneous segments. The segmentation step is followed by classifying each identified segment into a specific audio class. The second category consists of systems that classify short-term frames into audio classes and combine those predictions into homogenous segments using post-processing methods like Viterbi decoding or median filtering.

The most popular audio features among top submissions of the MIREX SMC challenge (2015 edition) were Root Mean Squared Energy, Spectral features, ZCR, Mel-Frequency Cepstral Coefficients (MFCC), Mel-scaled and regular Spectrograms, and Chroma-based features. In addition to these, Constant-Q Transform spectrograms and Periodograms were also employed. In the 2018 edition, most submissions used the Mel spectrogram as a feature for classification. A few works also explored hidden-layer embeddings obtained from deep networks as features. The popular choices of classifiers in MIREX 2015 were heuristic-based decision functions, Logistic Regression, Random Forests, Support Vector Machines (SVM), Restricted Boltzmann Machines, and shallow Convolutional Neural Networks (CNN). However, in the 2018 edition of MIREX, most submissions preferred deep CNNs as the classifier. A few authors also employed Recurrent Neural Networks, Deep Residual Networks, and

Multi-Layer Perceptrons.

Most existing SMC works employ features derived from the magnitude spectrum of audio signals. In comparison, phase information is rarely used for this particular task. Audio signals have frequency-encoded information where magnitude plays a more critical role than phase [237]. However, speech and music signals possess appealing properties that motivated the exploration of phase-based features in this task. Phase is believed to carry the information about the shape of signal waveform [161]. The waveform of speech and music signals can be very different. Also, the temporal evolution of instantaneous frequencies that model various musical phenomena like vibrato is characterized by time warping [238].

An audio signal is described as the superposition of harmonic components with global amplitude modulation and time warping (or phase variations) [238]. Such time-warping phenomena are absent in speech signals. Furthermore, music has higher temporal regularity than speech because music follows beat isochrony and a meter [239]. Such temporal properties might be captured in their phase information. Additionally, signal distortions are more pronounced in the corresponding group delay time than in the phase response [240]. The human auditory system is more sensitive to high-frequency signal components lagging behind low-frequency components [240]. Speech is essentially a low-frequency signal when compared to music. Hence, delays between high and low-frequency components of music might be more pronounced than in speech. The human auditory system is sensitive to phase distortion of speech and music signals when the maximum difference in group delays at the high (near 8 kHz) and low (in the neighborhood of 125 Hz) frequencies is above 50-70 ms [241]. More specifically, distortions are noticeable when maximum differences between group delay times for low (125 Hz) and high (8 kHz) frequencies are 40 ms and 80 ms for speech and music, respectively [240]. Moreover, phase distortions of speech signals are perceived as more substantial than those of music [240]. Such a difference between the signals may be exploited for their discrimination.

Mukherjee et al. [242] state that phase synchronization is a quantification tool in communication theory to establish similarity between two music signals. It can be argued that the non-synchronous phase of speech and music signals may help discriminate between them. Another discriminating aspect between speech and music is the number of sound sources in these signals. Generally, multiple speakers simultaneously speaking are relatively less frequent. In other words, most of the available speech data can be considered to be produced by a single system at a time. Most music signals in real-

life scenarios involve multiple systems producing different tones (polyphonic). Hence, most music data may be assumed to be produced by multiple systems playing simultaneously. The phase relationship between the sound components of speech (mostly single-system) must differ from music signals (mostly multi-system). Motivated by these ideas, the present chapter explores the phase information for discriminating between speech and music. However, we believe that the phase information of the signals involved in Multi-Speaker Speech (MSS) and Single Instrument Monophonic music (SIM) may not be sufficient to identify their audio class. We assume that the availability of MSS and SIM in real-life scenarios must be relatively scarce and can be ignored for this study.

This chapter investigates the importance of phase information in the current task by exploring three existing phase-based features previously employed in other speech processing applications. These features are MFCC computed from the Hilbert envelope of the Numerator of Group Delay spectrum (HNGDCC) [243], Modified Group Delay Cepstral Coefficient (MGDCC) [244], and Instantaneous Frequency Cosine Coefficient (IFCC) [245]. The relevance of these features and their respective contributions to the current task are described in section 4.2. The current proposal of using phase-based features is benchmarked against four state-of-art baseline magnitude-based approaches and validated on six datasets. The principal contributions of this chapter are listed next.

(i) This chapter explores phase information in the SMC task. In this regard, three existing phase-based audio features are explored, viz., Mel-Frequency Cepstral Coefficients computed from Hilbert Envelope of Numerator of Group Delay spectrum, Modified Group Delay Cepstral Coefficients, and Instantaneous Frequency Cosine Coefficients.

(ii) Canonical Correlation Analysis (CCA), Multivariate ANalysis Of VAriance (MANOVA), and generalization capability of the phase-based features across datasets are investigated and reported.

(iii) This chapter analyzes the SMC performance of phase-based features when the test signals are corrupted with noise at various SNR levels.

(iv) A preliminary study is also provided to show the effectiveness of the phase-based features in combination with the magnitude-based ones in segmenting continuous sequences of speech and music.

The organization of this chapter is as follows. Section 4.2 describes the three phase-based features

used in this chapter and their importance in the current context. Brief descriptions of the baseline features used to benchmark the current proposal are provided in subsection 4.2.4. Experiments and results are discussed in section 4.4. Finally, the chapter is summarized in section 4.5.

4.2 Features

This proposal aims to validate that SMC performance can be improved by utilizing valuable information from the signal's phase component. Three existing phase-based features, viz., HNGDCC, MGDCC, and IFCC, are explored in this context. The performance of these phase-based features is compared with four baseline magnitude-based features. An overview of the proposed SMC system is described next.

The proposed system performs the following major steps. First, a given signal $x[n]$ $(n = 0, \ldots (N_s - 1))$ that is sampled at sr Hz, is split into L short-term audio frames x_l $(l = 0, \ldots (L-1))$. Here, x_l is t_w milliseconds (ms) long and starts with a shift of t_s ms after x_{l-1}. Second, the phase-based and baseline features are computed and stored for each x_l. Let, F_l be a feature vector computed for frame x_l $(F_l \in \mathbb{R}^D)$. Third, a classifier model is trained to perform frame-level classification. This chapter uses Gaussian naive Bayes, SVM, Deep Neural Network (DNN), and CNN as classifiers. A context of $2W$ frames around every F_l is given as input to the classifier, which learns to predict the class of F_l. The mean $F_\mu^{(l)}$ and standard deviation $F_\sigma^{(l)}$ of the $2W$ frames around F_l are concatenated to form the input vector $V_l = \left[F_\mu^{(l)}, F_\sigma^{(l)}\right]$ in the case of naive Bayes, SVM and DNN classifiers. Here, the mean and standard deviation of the d^{th} feature dimension is computed as equation 4.1 and equation 4.2.

$$F_\mu^{(l)}[d] = \frac{1}{2W} \sum_{i=l-W}^{l+W-1} F_i[d] \tag{4.1}$$

$$F_\sigma^{(l)}[d] = \left| \sqrt{\frac{1}{2W} \sum_{i=l-W}^{l+W-1} \left(F_i[d] - F_\mu^{(l)}[d]\right)^2} \right| \tag{4.2}$$

Here, $d = 0, \ldots (D-1)$. In the case of CNN classifier, the input is a 2-dimensional feature-patch S_l of size $d \times 2W$ which is extracted as $S_l = \left[F_{l-W}^T, \ldots F_l^T, \ldots F_{l+W}^T\right]$.

Finally, the trained classifiers are used to predict the audio class of the center frame in each $2W$-sized context window extracted from the test data. The following subsections describe the computation process of the phase-based features and their relevance in the current task. Baseline features used for

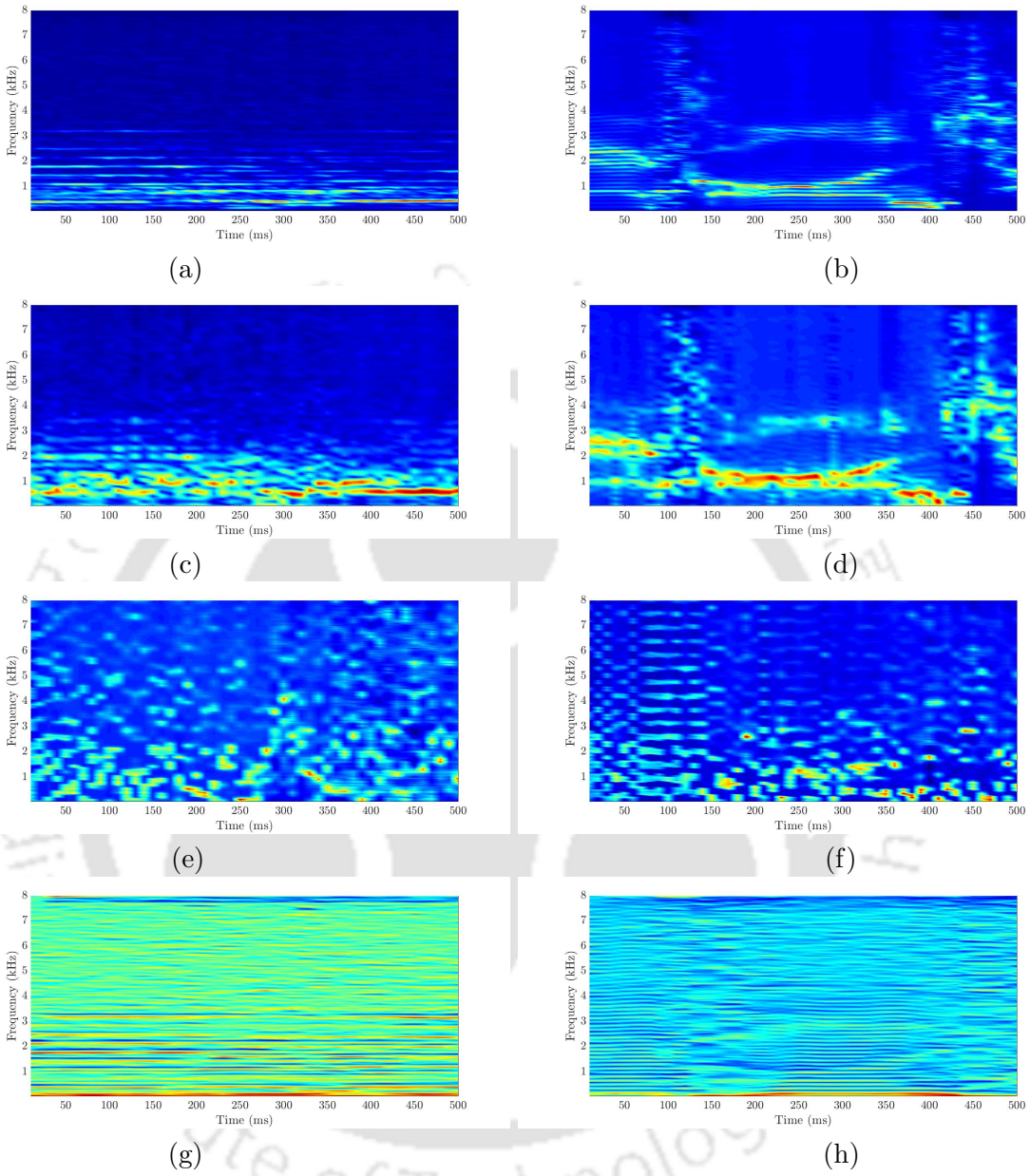


Figure 4.1: Illustrating the difference between time-frequency representations generated from the magnitude and phase components of **music** and **speech** signals. Sub-figures show (a) DFT spectrogram of music, (b) DFT spectrogram of speech, (c) HNGD spectrogram of music, (d) HNGD spectrogram of speech, (e) MGD spectrogram of music, (f) MGD spectrogram of speech, and (g) IFQ spectrogram of music, (h) IFQ spectrogram of speech.

performance comparison in this chapter are briefly described later.

4.2.1 Mel-frequency Cepstral Coefficients of Hilbert Envelope of the Numerator of Group Delay

The Discrete Fourier Transform (DFT) spectrograms and HNGD spectrograms for speech and music signals are shown in Fig. 4.1(a)-(d). It was argued in [243] that the HNGD spectrum captures the formant peaks better than the DFT spectrum. It can be observed in Fig. 4.1(c) and Fig. 4.1(d) that the HNGD spectrogram slightly smoothes out the harmonics. For a speech signal, the HNGD algorithm enhances formants. The formant structure present in speech is a unique characteristic that distinguishes it from music [229]. A feature that captures the formant information better can be expected to perform well in SMC. Thus, MFCC computed from the HNGD spectrum (HNGDCC, used as phase-feature $P1$) is explored in this chapter.

The HNGD spectrum was proposed by Yegnanarayana et al. [243] to deal with the smoothing-out of formant peaks in the DFT spectrum. This spectrum is extracted from the group delay function representing the underlying signal's phase component. Previously, Murthy et al. [218] proposed a method for extracting epoch locations in the speech by filtering the signal with a zero-frequency resonator. The authors showed that this epoch extraction approach does not smear the discontinuities in the time-domain signal. A similar approach in the time domain, called zero-time windowing, is used to compute the HNGD spectrum.

Let $x[n]$ be an audio segment that is differenced to remove the DC bias. A zero-time windowing of the l^{th} short-term frame x_l of size $2N_f$ obtained from $x[n]$ is performed using a window function $w[m]$ $(m = 0, \ldots (2N_f - 1))$ to obtain $x_l^z = x_l \cdot w$. Here, w is defined by equation 4.3.

$$w[m] = \begin{cases} 0, & m = 0 \\ \dfrac{1}{4sin^2(\frac{\pi m}{4N_f})}, & m = 1, \ldots 2N_f - 1 \end{cases} \tag{4.3}$$

Yegnanarayana [246] showed that speech formants could be enhanced by exploiting the properties of group-delay function $\tau[k]$. The $\tau[k]$ is computed according to equation 4.4 [247].

$$\tau[k] = \frac{X_R[k]Y_R[k] + X_I[k]Y_I[k]}{X_R^2[k] + X_I^2[k]}, \; k = 0, \ldots 2N_f - 1 \tag{4.4}$$

Here, $X[k] = X_R[k] + jX_I[k]$ is the $2N_f$-point complex DFT of $x_l^z[m]$ and $Y[k] = Y_R[k] + jY_I[k]$ is the $2N_f$-point complex DFT of $y[m] = m \cdot x_l^z[m]$. Since the input signals are real, only N_f components are retained in $\tau[k]$. The numerator of $\tau[k]$ is the Numerator of Group Delay (NGD). Compared to

$\tau[k]$, NGD has a higher resolution around formants [246,248]. Hence, only NGD is retained for further processing. The NGD function is sign-reversed and double-differenced (referred to as DD-NGD). Since NGD is a function of frequency, the double-differencing operation to obtain DD-NGD is performed in frequency. However, these operations may affect the orientation of higher-order resonances [243]. Moreover, spectral valleys may affect nearby peaks and increase the difficulty of identifying peak locations. Hence, the Hilbert envelope of the DD-NGD spectrum (defined as the HNGD spectrum) is computed to highlight the peaks using equation 4.5.

$$HNGD[k] = \sqrt{DD - NGD^2[k] + DD - NGD_h^2[k]} \tag{4.5}$$

Here, $k = 0, \ldots (N_f - 1)$ and $DD - NGD_h[k]$ is the Hilbert transform of $DD - NGD[k]$. The $DD - NGD_h[k]$ function is computed using equation 4.6.

$$DD - NGD_h[k] = \mathcal{F}_{\mathcal{D}}^{-1}\{\mathbf{D}_h[\tilde{k}]\} \tag{4.6}$$

Here, $\tilde{k} = 0, \ldots (N_f - 1)$ and $\mathcal{F}_{\mathcal{D}}^{-1}$ is the inverse DFT operator. $\mathbf{D}_h[\tilde{k}]$ is defined according to equation 4.7.

$$\mathbf{D}_h[\tilde{k}] = \begin{cases} -j\mathbf{D}[\tilde{k}], & 0 < \tilde{k} \leq \dfrac{N_f}{2} \\ j\mathbf{D}[\tilde{k}], & \dfrac{N_f}{2} < \tilde{k} \leq (N_f - 1) \end{cases} \tag{4.7}$$

Here, $\mathbf{D}[\tilde{k}]$ is the N_f-point DFT of $DD - NGD[k]$. The reader is encouraged to refer to the original paper for a detailed treatment of HNGD computation and related issues [243].

Khonglah et al. [249] showed that MFCC computed from the HNGD spectrum performed better than that computed from the DFT spectrum for the task of clean speech vs. speech with background music classification. The MFCC computation process involves smoothing the HNGD spectrum using a series of n_{mel} triangular Mel-filters. The r^{th} Mel filter is defined according to equation 4.8 [250].

$$\mathbf{H}_r[k] = \begin{cases} 0, & \text{if } f_k < \tilde{f}_{r-1} \\ \dfrac{f_k - \tilde{f}_{r-1}}{\tilde{f}_r - \tilde{f}_{r-1}}, & \text{if } \tilde{f}_{r-1} \leq f_k < \tilde{f}_r \\ \dfrac{f_k - \tilde{f}_{r+1}}{\tilde{f}_r - \tilde{f}_{r+1}}, & \text{if } \tilde{f}_r \leq f_k < \tilde{f}_{r+1} \\ 0, & \text{if } f_k \geq \tilde{f}_{r+1} \end{cases} \tag{4.8}$$

Here, $r = 0, \ldots (n_{mel} - 1)$, $k = 0, \ldots (N_f - 1)$, $n_{mel} << N_f$, and $\tilde{f}_r$ are the center-frequencies of n_{mel} Mel-filters. Hertz-scale frequencies f_k are converted to Mel-scale frequencies ϕ_k using equation 4.9.

$$\phi_k = 2595 \log_{10} \left(1 + \frac{f_k}{700}\right) \tag{4.9}$$

Next, n_{mel} equally spaced frequencies in the Mel-scale are reconverted to Hertz-scale and selected as center-frequencies for the Mel filter-bank. Typically, the center frequencies are selected within a predetermined frequency range. Subsequently, the natural logarithm of the sum of element-wise product between Mel filters $\mathbf{H}$ and the DFT spectrum $\mathbf{X}$ is computed according to equation 4.10.

$$\hat{\mathbf{X}}\left[r\right] = \ln \left(\sum_{k=0}^{(N_f-1)} \left|\mathbf{X}[k]\right| \cdot \mathbf{H}_r\left[k\right] \right) \tag{4.10}$$

The Mel filter-bank energies $\hat{\mathbf{X}}$ are then decorrelated and compacted using discrete cosine transformation as equation 4.11 [250].

$$\psi[j] = \sum_{r=0}^{(n_{mel}-1)} \hat{\mathbf{X}}\left[r\right] \cos\left(l \frac{\pi}{n_{mel}} \left(r - \frac{1}{2}\right)\right) \tag{4.11}$$

Here, $j = 0, \ldots (n_m - 1)$, and ψ is the HNGDCC feature vector. It may be noted that the standard MFCC feature differs from HNGDCC. Standard MFCC is computed from the DFT magnitude spectrum, whereas HNGDCC is computed from the HNGD spectrum that contains phase information of the underlying signal. Finally, first 13 cepstrum coefficients ($[\psi[0], \psi[1], \ldots \psi[12]]$) are retained for each frame, along with their 13-Δ and 13-$\Delta\Delta$ coefficients together to form the 39-dimensional $P1$ feature.

4.2.2 Modified Group Delay Cepstral Coefficient

The group delay function and phase spectrum of a signal have similar characteristics. Murthy et al. [244] observed that the standard group delay function might contain unwanted spikes due to pitch peaks, noise, and window effects, thereby making it ill-defined. Hence, they proposed a less noisy Modified Group Delay (MGD) function that better preserves the finer structure. The formulation of the original group-delay function is provided in equation 4.4. Murthy et al. [244] proposed to compute the MGD function $\tau_{MGD}[k]$ by modifying equation 4.4 to equation 4.12.

$$\tau_{MGD}[k] = sgn \cdot \left| \frac{X_R[k]Y_R[k] + X_I[k]Y_I[k]}{S[k]^{2\gamma}} \right|^{\rho} \tag{4.12}$$

Here, $X[k] = X_R[k] + jX_I[k]$ and $Y[k] = Y_R[k] + jY_I[k]$ are the respective $2N_f$-point complex DFTs of $x_l[m]$ and $y[m] = m \cdot x_l[m]$, $S[k]$ is the smoothed version of $|X[k]|$, $|\cdot|$ is the absolute value function, and sgn represents the sign of $\frac{X_R[k]Y_R[k] + X_I[k]Y_I[k]}{S[k]^2}$. The parameters ρ and γ can be tuned to reduce the spiky nature of the function. The MGD function is converted to cepstral features using the Discrete Cosine Transform (DCT) (equation 4.11). A detailed description of the computation of the MGD spectrum can be found in [244]. The first 13 DCT coefficients in each frame, along with 13-Δ and 13-$\Delta\Delta$, taken together form the 39-dimensional MGDCC feature.

The time-frequency representation of MGD computed for music and speech signals are shown in Fig. 4.1(e) and Fig. 4.1(f), respectively. MGD is described as the frequency derivative of phase component and is known to capture the inter-formant information in the speech spectra [244]. The lack of a strict formant structure in music makes MGD a decent choice for use in the discrimination of music and speech signals. Previously, the MGDCC has been successfully used in speech phoneme recognition [244]. Therefore, this chapter considers MGDCC (phase feature $P2$) the second phase-based feature for SMC.

4.2.3 Instantaneous Frequency Cosine Coefficient

Vijayan et al. [245] proposed the IFCC as a feature computed from the analytic phase of speech signals. They employed the IFCC feature in the task of speaker verification. Authors employed properties of the Fourier transform to avoid explicit computation of the analytic phase and overcome the issue of phase wrapping. The input signal is first passed through a bank of C narrow-band (NB) triangular-shaped filters, linearly spaced with 50% overlap of frequency bins. The NB components thus obtained can be added to reconstruct the original wide-band signal. The analytic signal for each of these NB components is computed next. An analytic signal representation $z[c, n]$ of the c^{th} NB component $g[c, n]$ is defined according to equation 4.13.

$$z[c, n] = \sqrt{g^2[c, n] + g_h^2[c, n]} \tag{4.13}$$

Here, $c = 0, \ldots (C - 1)$, $n = 0, \ldots (N_s - 1)$, $g_h[c, n]$ is the Hilbert transform of $g[c, n]$, computed as shown in equation 4.7 and N_s is length of the signal. The derivative of the unwrapped analytic phase is referred to as Instantaneous FreQuency (IFQ) [245]. Vijayan et al. [245] proposed that IFQ can be computed without being affected by phase wrapping. The derivative $z'[c, n]$ of c^{th} NB analytic signal

$z[c,n]$ can be computed using differentiation property of the Fourier transform as equation 4.14 [247]:

$$z'[c,n] = j\mathcal{F}_{\mathcal{D}}^{-1} k \cdot Z[c,l] \tag{4.14}$$

Here, $l = 0, \ldots (N_s - 1)$, $\mathcal{F}_{\mathcal{D}}^{-1}$ denotes the inverse DFT, N_s is length of the NB signal and $Z[c,l]$ is the DFT of the analytic signal $z[c,n]$. Subsequently, the IFQ $\theta[c,n]$ of the c^{th} discrete-time NB component is computed using equation 4.15.

$$\theta[c,n] = \frac{2\pi}{N_s} Re \left\{ \frac{\mathcal{F}_{\mathcal{D}}^{-1}(l \cdot Z[c,l])}{\mathcal{F}_{\mathcal{D}}^{-1}(Z[c,l])} \right\} \tag{4.15}$$

Here, $n = 0, \ldots (N_s - 1)$. The process of obtaining the analytic signal from the NB signal is described in [251]. The C IFQ sequences are split into overlapping short-term frames after computing the IFQ for all the NB components of a given signal. Let, a short-term frame of the c^{th} NB component be represented as $\theta_l[c,m]$, where $m = 0, \ldots (N-1)$, $l = 0, \ldots (L-1)$ and N is the window size. The IFQ values in each c^{th} θ_l are averaged to obtain a C-dimensional mean IFQ vector (equation 4.16).

$$IFQ_l[c] = \frac{1}{N} \sum_{m=0}^{(N-1)} \theta_l[c,m] \tag{4.16}$$

Finally, each C-dimensional short-term mean IFQ frame feature vector is converted into a compact representation (IFCC) by performing a DCT. The first 13 DCT coefficients in each frame are retained, along with 13-Δ and 13-$\Delta\Delta$ coefficients to form a 39-dimensional IFCC feature.

The instantaneous frequency of a signal depicts the rate of change of the unwrapped analytic phase [245]. These changes can be observed as horizontal striations in the IFQ spectrograms (Fig. 4.1(g) and Fig. 4.1(h)). These horizontal striations are the temporal evolution of harmonics in the signal. Harmonics in speech are damped faster than music because the speech production system is a damped system [221]. Thus the IFQ spectrogram can capture discriminating information between the two classes. Hence, compact IFQ information in the form of IFCC (phase feature $P3$) is selected in this chapter as the third phase feature.

4.2.4 Baseline features for performance comparison

The effectiveness of phase-based features in the current task can be judged when their performance is compared with standard features computed from the magnitude spectrum of audio signals. Hence, the phase-based features are benchmarked against the following four state-of-the-art magnitude-based

feature sets.

The group of features used by Khonglah et al. [148] ($B1$) in the SMC task is considered the first baseline. The set $B1$ includes features primarily used to model speech signals. One such feature is the Normalized Autocorrelation Peak Strength (NAPS) of the Zero-Frequency Filtered Signal (ZFFS). The authors explain that NAPS of ZFFS exploits an apriori knowledge of human pitch range that is distinct from music. Peak to Side-lobe Ratio (PSR) of the Hilbert envelope of the Linear Prediction Residual signal is another component feature of $B1$. The PSR feature models impulse-like excitations in speech signals that differ from music. The baseline $B1$ includes features like log Mel-spectrum energy in the low-frequency range, 4Hz modulation spectrum energy, spectral flux, spectral centroid, spectral roll-off, Zero Crossing Rate (ZCR) and percentage of low energy frames.

The group of features used by Sell et al. [140] ($B2$) is considered the second baseline. The set $B2$ includes two chroma-based features proposed by the authors. Here, the authors proposed chroma difference and chroma high-frequency to model music tonality. The chroma features measure pronounced peakiness inherent in the spectra of music signals. The baseline $B2$ further includes the standard deviation of normalized root mean square, silent interval ratio and frequency, ZCR variance, spectral centroid and flux variance, Mel-frequency subband modulation syllabic rate energy, and spectral centroid.

Third, MFCC ($B3$) computed from the DFT magnitude spectrum is one of the most widely used magnitude-based features in various speech processing applications, including SMC [149]. The MFCCs are mainly used to model the vocal tract system information. The lack of a vocal tract-like system in the music production process makes MFCC an outstanding feature to distinguish between the two audio classes. Thus, this feature has been used as the third baseline in this chapter.

Fourth, Mel-spectrogram ($B4$) was used as a feature by Doukhan et al. [2] in their winning submission to MIREX 2018 speech/music detection challenge. A Mel filter-bank scales the audio spectrum along the human auditory perception scale that helps discriminate between speech and music. Hence, $B4$ has been considered the fourth baseline feature in this chapter.

4.2.5 Feature computation parameters

The phase-based and baseline features have all been computed for $t_w = 25$ms and $t_s = 10$ms. The audio signals have been resampled to $sr = 16000$Hz. A given audio signal $x[n]$ is first pre-emphasized using a first-order auto-regressive filter as $x[n] = \tilde{x}[n] - C \cdot \tilde{x}[n-1]$, before computing the phase and

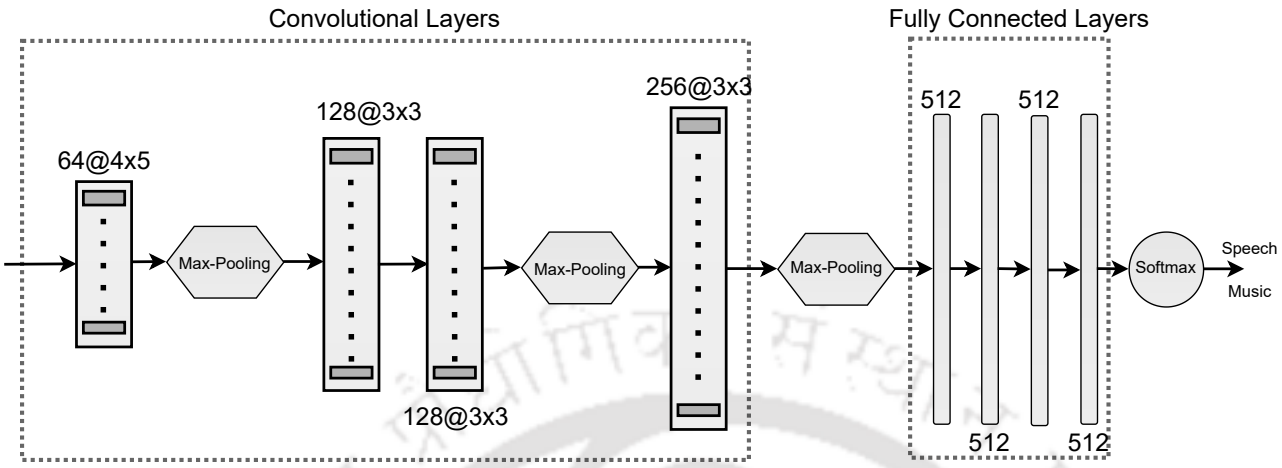


Figure 4.2: The architecture of the CNN classifier (proposed in [2]) used for performing speech vs. music classification in this chapter.

magnitude-based features. Here, $C = 0.97$ is the preemphasis factor, $\tilde{x}[n]$ is the audio signal before pre-emphasis, $n = 0, \ldots (N_s - 1)$ and N_s is the number of samples in $\tilde{x}[n]$. The baseline features $B3$, $B4$, and the phase-based feature $P1$ have been computed using a 21-band Mel filter-bank in the frequency range of (20Hz, 8000Hz). The features $B3$, $P1$, $P2$ and $P3$ have been computed as 39-dimensional features composed of 13-cepstral, 13-Δ and 13-$\Delta\Delta$ coefficients. The Δ window is set as 9 frames. The values of ρ and γ used in the computation of the $P2$ feature have been obtained after performing a grid-search on the GTZAN [204] dataset using all combinations of values over $[0.1, 0.2, \ldots 1.0]$ for both the parameters. The best performing values were found to be $\rho = 0.1$ and $\gamma = 0.3$. The different audio preprocessing and feature extraction steps have been performed using available functions in the python library *Librosa* [252].

4.3 Classifiers

The SMC results are reported using naive Bayes, SVM, DNN, and CNN classifiers. The naive Bayes and SVM classifiers are implemented using the *Scikit-learn* library [253]. The DNN and CNN classifiers are implemented using the *Tensorflow* library [254]. The naive Bayes classifier learns Gaussian distributions over features of each class and performs Maximum A Posteriori classification. The SVM classifier is used with a Radial Basis Function kernel. The cost and RBF kernel bandwidth parameters of the SVM classifier for each feature are tuned for every cross-validation fold. The cost and RBF kernel bandwidth parameters of the SVM classifier are varied in the range $[2^{-5}, 2^{-4}, \ldots 2^{4}]$

Table 4.1: Results of grid-search for optimal **DNN** architecture over *Movie-MUSNOMIX* dataset, reported as $\mu \pm \sigma$ of average $F1$-scores. The final results reported in the chapter have been computed using the parameters ranked I.

Feature	Rank	Hidden layers	Hidden Nodes	Avg. $F1$-score ($\mu \pm \sigma$)
$B1$	I	5	250	89.26 ±0.48
	II	3	250	89.2 ±0.97
	III	4	500	89.19 ±0.73
$B2$	I	3	1000	87.08 ±1.14
	II	2	500	87 ±2.25
	III	4	500	86.98 ±0.89
$B3$	I	4	500	89.58 ±0.80
	II	3	500	89.29 ±1.15
	III	2	250	89.19 ±0.69
$B4$	I	2	1000	90.2 ±0.54
	II	2	500	89.97 ±0.79
	III	2	250	89.86 ±0.68
$P1$	I	3	500	76.49 ±0.91
	II	2	500	76.15 ±1.61
	III	4	1000	75.99 ±1.19
$P2$	I	2	1000	75.43 ±0.52
	II	3	1000	75.43 ±1.13
	III	3	250	75.42 ±0.62
$P3$	I	5	500	73.49 ±0.80
	II	2	500	73.21 ±0.38
	III	4	500	73.2 ±0.11

to obtain their optimal values by using grid-search.

The DNN architecture is finalized by performing a grid search. This search is performed by varying the number of hidden layers in the range $[2, \ldots 5]$. The number of hidden layer neurons is varied in $[100, 250, 500, 1000]$. The results of grid-search are presented in Table 4.1 and in Table 4.2 for the *Movie-MUSNOMIX* and *MUSAN* datasets, respectively. This chapter uses the best-performing DNN architecture parameters for further experiments. The finalized architecture has the same number of neurons for all the hidden layers. The hidden layers use $ReLU$ activation, while the output layer with a single neuron uses *Sigmoid* activation. The DNN is trained using Adam [231] optimizer for minimizing a binary cross-entropy loss function. Input to the DNN classifier is a concatenated vector

Table 4.2: Results of grid-search for optimal **DNN** architecture over *MUSAN* dataset, reported as $\mu \pm \sigma$ of average $F1$-scores. The final results reported in the chapter have been computed using the parameters ranked I.

Feature	Rank	Hidden layers	Hidden Nodes	Avg. $F1$-score ($\mu \pm \sigma$)
$B1$	I	3	250	88.73 ±1.37
	II	2	250	88.67 ±0.67
	III	4	250	88.34 ±1.09
$B2$	I	2	100	85.76 ±1.80
	II	2	250	84.98 ±0.65
	III	5	500	84.97 ±1.54
$B3$	I	3	100	94.72 ±0.37
	II	2	100	94.33 ±0.45
	III	5	100	94.24 ±0.82
$B4$	I	5	250	81.82 ±1.25
	II	4	500	81.65 ±2.77
	III	5	1000	81.59 ±1.68
$P1$	I	5	100	86.46 ±1.05
	II	2	250	86.44 ±0.72
	III	3	500	86.39 ±0.45
$P2$	I	4	500	67.66 ±2.26
	II	4	100	67.07 ±8.29
	III	4	250	64.74 ±5.05
$P3$	I	4	100	85.21 ±1.06
	II	5	100	85.08 ±0.37
	III	4	250	85.08 ±1.59

of the mean and standard deviation of frame features in a context window of $W = 34$ (see section 4.2).

The CNN architecture proposed by Doukhan et al. [2] was adjudged best among MIREX 2018 submissions. This CNN architecture is adopted in the present chapter. The CNN consists of 4 convolutional layers with 64, 128, 128, and 256 convolution kernels, respectively. The convolution layers are followed by 4 fully connected layers with 512 neurons each. The model performs Max-Pooling operation after the 1^{st}, 3^{rd} and 4^{th} convolutional layers. Hidden-layers use the $ReLU$ activation function. The sizes of (3×3) convolutional filters are kept the same as in the original paper. The CNN is trained using Adam [231] optimizer with a binary cross-entropy loss function. A block of features in a context window of $W = 34$ is used as input to the CNN classifier (see section 4.2).

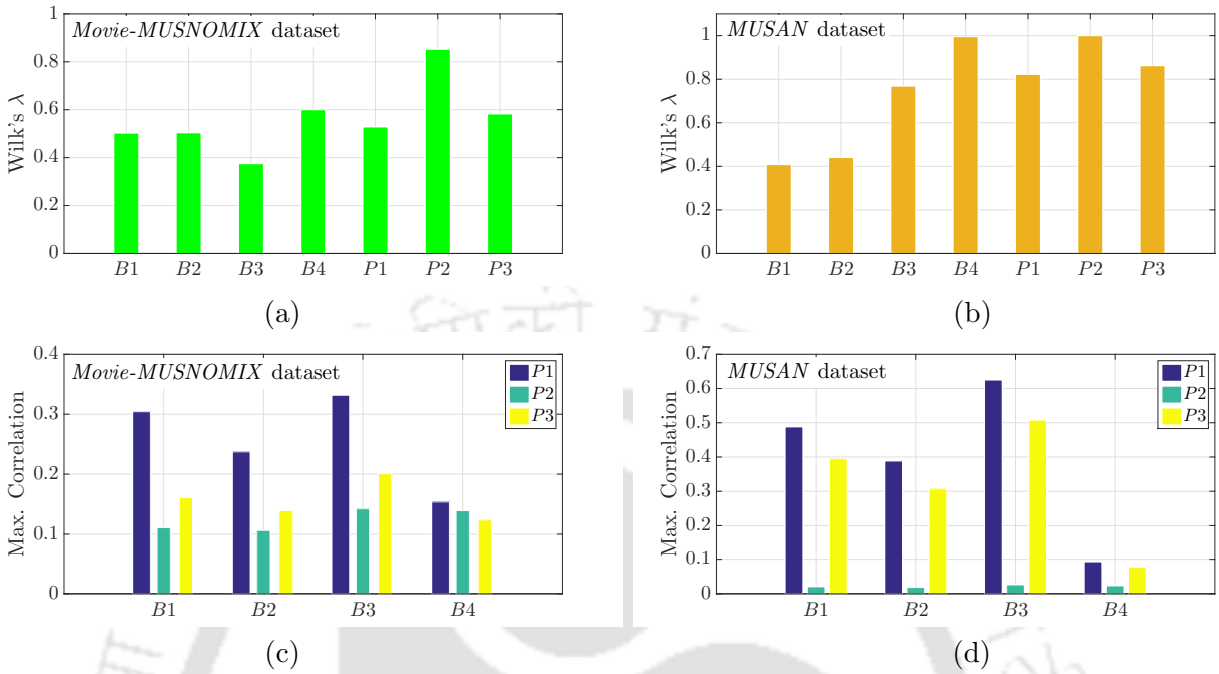


Figure 4.3: Bar charts in the top row illustrated the statistical significance of baseline and phase-based features in terms of Wilks' Lambda values obtained by performing MANOVA over (a) *Movie-MUSNOMIX* and (b) *MUSAN* datasets. A feature set with a lower value of Λ is preferred. In the bottom row, bar charts illustrate the maximum CCA values between every pair of baseline and phase-based features for (c) *Movie-MUSNOMIX* and (d) *MUSAN* datasets, respectively. For both the datasets, phase-based features have a low correlation with the baseline features.

4.4 Evaluation

The classification performance of the proposed system is benchmarked on three public datasets – (a) *GTZAN* Music/Speech collection [204] (approximately 1 hour), (b) *Scheirer-Slaney* Music-Speech corpus [99] (approximately 1 hour), and (c) *MUSAN*-A Music, Speech and Noise corpus [203] (approximately 102 hours). Results are also reported for the contributed Movie-MUSNOMIX dataset (approximately 3 hours and 40 minutes of speech and music signals). The segmentation of continuous speech and music intervals is studied using the *Muspeak* [205] and *DAFx-12* [206] datasets. This chapter aims to validate that SMC performance can be improved by utilizing valuable information from the signal's phase component. In this context, three existing phase-based features, viz. HNGDCC ($P1$), MGDCC ($P2$), and IFCC ($P3$) are explored. Performances of the phase-based features are compared with four baseline magnitude-based features. The results are reported using mean (μ) and standard deviation (σ) of average $F1$-scores computed using 3-fold cross-validation. Source codes of

Table 4.3: Illustrating the frame-level classification performances of phased-based and baseline feature sets for the *GTZAN* and *Scheirer-Slaney* datasets computed using **naive Bayes** and **SVM** classifiers. Results are reported as "mean (μ) $F1$-score (over 3-fold cross-validation) $\pm$ standard deviation (σ)" of all-class average classification performance. The mean and standard deviation of $F1$-scores are expressed in percentage. Feature combinations that improve upon the baseline are highlighted with ▲.

		GTZAN		*Scheirer-Slaney*	
	Features	**naive Bayes** ($\mu \pm \sigma$)	**SVM** ($\mu \pm \sigma$)	**naive Bayes** ($\mu \pm \sigma$)	**SVM** ($\mu \pm \sigma$)
Baseline Features	$B1$	68.63 ±6.41	80.82 ±4.23	71.39 ±0.23	85.17 ±0.46
	$B2$	69.15 ±5.39	78.41 ±2.56	69.15 ±0.47	81.38 ±0.30
	$B3$	74.12 ±7.68	85.68 ±5.14	76.37 ±1.77	86.36 ±2.82
	$B4$	70.77 ±3.66	82.87 ±2.12	77.82 ±3.80	85.73 ±1.55
Phase Features	$P1$	64.63 ±4.95	72.98 ±2.12	61.11 ±1.63	70.04 ±2.10
	$P2$	51.79 ±0.68	59.85 ±0.73	52.11 ±2.24	63.51 ±1.17
	$P3$	64.24 ±2.47	70.82 ±1.68	58.07 ±3.17	62.75 ±2.73
All Combinations	$B1$+$B2$+$B3$+$B4$ ($\mathbf{B}_{All}$)	76.39 ±2.70	90.73 ±1.92	82.35 ±1.81	91.67 ±0.99
	$P1$+$P2$+$P3$ ($\mathbf{P}_{All}$)	66.43 ±2.42	74.81 ±1.52	61.01 ±1.79	67.08 ±3.78
	$\mathbf{B}_{All}$+$P1$	77.73 ±7.06 ▲	91.30 ±3.25 ▲	80.69 ±2.01	93.07 ±1.22 ▲
	$\mathbf{B}_{All}$+$P2$	76.62 ±3.95 ▲	90.87 ±2.87 ▲	80.10 ±1.85	92.97 ±1.20 ▲
	$\mathbf{B}_{All}$+$P3$	78.30 ±6.91 ▲	91.31 ±3.09 ▲	80.69 ±2.14	92.58 ±1.22 ▲
	$\mathbf{B}_{All}$+($P2$+$P3$)	77.00 ±1.56 ▲	89.84 ±2.02	80.55 ±1.74	87.19 ±2.04
	$\mathbf{B}_{All}$+($P1$+$P3$)	78.38 ±4.63 ▲	90.88 ±1.97 ▲	80.92 ±2.41	88.03 ±2.12
	$\mathbf{B}_{All}$+($P1$+$P2$)	75.14 ±1.16	89.94 ±1.93	79.65 ±2.09	88.18 ±1.80
	$\mathbf{B}_{All}$+$\mathbf{P}_{All}$	78.19 ±5.96 ▲	91.32 ±2.60 ▲	79.23 ±2.38	90.49 ±1.91

the present implementation are available online [1]. The statistical significance of the phase-based and magnitude-based features are analyzed next.

4.4.1 MANOVA and CCA

The statistical significance of features is studied in this chapter using MANOVA (see subsection 3.4.1) [232]. A high Λ value verifies the null hypothesis. A feature set is statistically significant for a classification task if it has a low Λ value (refutes the null hypothesis). The Λ values of baseline and phase-based feature sets are shown in the form of bar charts in Fig. 4.3. The baseline features

[1] Source codes: `https://github.com/mrinmoy-iitg/SMC_Phase_Features`

Table 4.4: Illustrating the frame-level classification performances of phase-based and baseline feature sets for the *MUSAN*, and *Movie-MUSNOMIX* datasets computed using **DNN** and **CNN** classifiers. Results are reported as "mean (μ) $F1$-score (over 3-fold cross-validation) $\pm$ standard deviation (σ)" of average classification performance. The mean and standard deviation of $F1$-scores are expressed in percentage. Feature combinations that improve upon the baseline are highlighted with ▲.

			MUSAN		*Movie-MUSNOMIX*	
		Features	**DNN** ($\mu \pm \sigma$)	**CNN** ($\mu \pm \sigma$)	**DNN** ($\mu \pm \sigma$)	**CNN** ($\mu \pm \sigma$)
Baseline	**Features**	$B1$	88.29 ±1.17	96.29 ±1.20	89.24 ±0.52	90.55 ±3.03
		$B2$	86.32 ±1.17	97.00 ±0.61	87.10 ±1.19	88.64 ±3.72
		$B3$	94.71 ±0.38	97.71 ±0.69	89.56 ±0.81	96.27 ±0.99
		$B4$	79.15 ±5.33	95.99 ±0.98	90.14 ±0.50	95.01 ±2.02
Phase	**Features**	$P1$	86.46 ±1.05	93.45 ±0.80	76.54 ±1.07	90.92 ±2.51
		$P2$	75.60 ±0.13	83.56 ±1.31	75.34 ±0.53	94.39 ±0.31
		$P3$	86.49 ±1.24	88.08 ±0.57	73.52 ±0.84	78.67 ±0.94
All	**Combinations**	$B1$+$B2$+$B3$+$B4$ ($\mathbf{B}_{All}$)	97.53 ±0.30	99.81 ±0.05	96.39 ±0.27	99.42 ±0.04
		$P1$+$P2$+$P3$ ($\mathbf{P}_{All}$)	90.73 ±0.97	95.90 ±1.25	84.26 ±0.26	94.97 ±0.76
		$\mathbf{B}_{All}$+$P1$	97.64 ±0.45 ▲	99.79 ±0.04	96.25 ±0.27	98.84 ±0.21
		$\mathbf{B}_{All}$+$P2$	97.90 ±0.21 ▲	99.83 ±0.07 ▲	96.26 ±0.14	99.15 ±0.20
		$\mathbf{B}_{All}$+$P3$	97.95 ±0.54 ▲	99.81 ±0.06	96.67 ±0.31 ▲	99.43 ±0.06 ▲
		$\mathbf{B}_{All}$+($P2$+$P3$)	98.18 ±0.39 ▲	99.87 ±0.03 ▲	96.49 ±0.01 ▲	94.97 ±0.76
		$\mathbf{B}_{All}$+($P1$+$P3$)	97.73 ±0.58 ▲	99.82 ±0.07 ▲	96.49 ±0.01 ▲	99.14 ±0.20
		$\mathbf{B}_{All}$+($P1$+$P2$)	97.89 ±0.29 ▲	99.84 ±0.02 ▲	96.55 ±0.30 ▲	98.96 ±0.09
		$\mathbf{B}_{All}$+$\mathbf{P}_{All}$	97.94 ±0.45 ▲	99.76 ±0.04	96.49 ±0.01 ▲	98.84 ±0.21

(except $B4$) have low Λ values for both datasets. Among phase-based features, $P2$ has the highest Λ value. Hence, $P2$ is the least statistically significant feature in the current task. $P1$ and $P3$ have comparatively better Λ values. Therefore, it may be expected that both $P1$ and $P3$ capture significant discriminative information that can be exploited in the task of SMC.

Canonical Correlation Analysis (CCA) [255] has been performed to determine whether the phase-based and baseline features can be combined to improve classification performance. A low correlation between a pair of feature sets suggests they carry complementary information. The maximum correlation computed between every pair of baseline and phase-based features over the *Movie-MUSNOMIX* and *MUSAN* datasets are shown in Fig. 4.3(c) and Fig. 4.3(d). Two observations can be drawn from

the correlation results. First, $P2$ has an extremely low correlation with all baseline features. As discussed earlier, $P2$ has low statistical significance. MGD spectrogram of music (see Fig. 4.1(e)) and speech (see Fig. 4.1(f)) do not seem to distinctly represent the tempo-spectral characteristics of the underlying signals. This effect might result in $P2$ having a low correlation with baseline features. Second, $B3$ and $B4$ seem to have the lowest correlation with the phase-based features. Thus, $B3$ and $B4$ may be the ideal candidates for feature combination [256].

4.4.2 Performance analysis

The performance of traditional naive Bayes and SVM classifiers using phase-based and magnitude-based features in SMC is tabulated in Table 4.3. The naive Bayes and SVM classifiers are employed for the smaller *GTZAN* and *Scheirer-Slaney* datasets. Expectedly, SVM performs better than the naive Bayes classifier across all individual features and combinations for both datasets. In general, individual performances of all the features suggest that the magnitude component carries more discriminative information between the classes. However, an ensemble of classifiers trained on all phase and magnitude-based features performs better than each feature's performance. A majority voting of the class labels predicted by each naive Bayes or SVM classifier in the ensemble is taken to obtain the final label. The performance of pair-wise feature combinations is not reported since at least 3 classifiers are required for majority voting. Ensemble of naive Bayes and SVM classifiers trained on all the magnitude and phase-based features ($\mathbf{B}_{All} + \mathbf{P}_{All}$) performs better than the combination of all magnitude-based features for the *GTZAN* dataset. Combining any one phase-based feature with all the magnitude-based features improves performance for both *GTZAN* and *Scheirer-Slaney* with both the classifiers. However, only for the *Scheirer-Slaney* dataset the ensemble of all classifiers trained on the magnitude and phase-based features did not provide the best results. This lack of performance improvement might be because the *Scheirer-Slaney* dataset is relatively more challenging. The *Scheirer-Slaney* dataset contains segments having speech overlapped with music segments [257]. These impure segments have quite different characteristics from pure speech and music.

The frame-level classification performances of the phase and magnitude-based features using DNN and CNN classifiers are tabulated in Table 4.4. The performances of DNN and CNN classifiers are not computed over the *GTZAN* and *Scheirer-Slaney* datasets because they are not large enough to train deep models. The individual performances of phase-based features are lower than that of the baseline features for both datasets. The popularity of magnitude-based features for SMC is

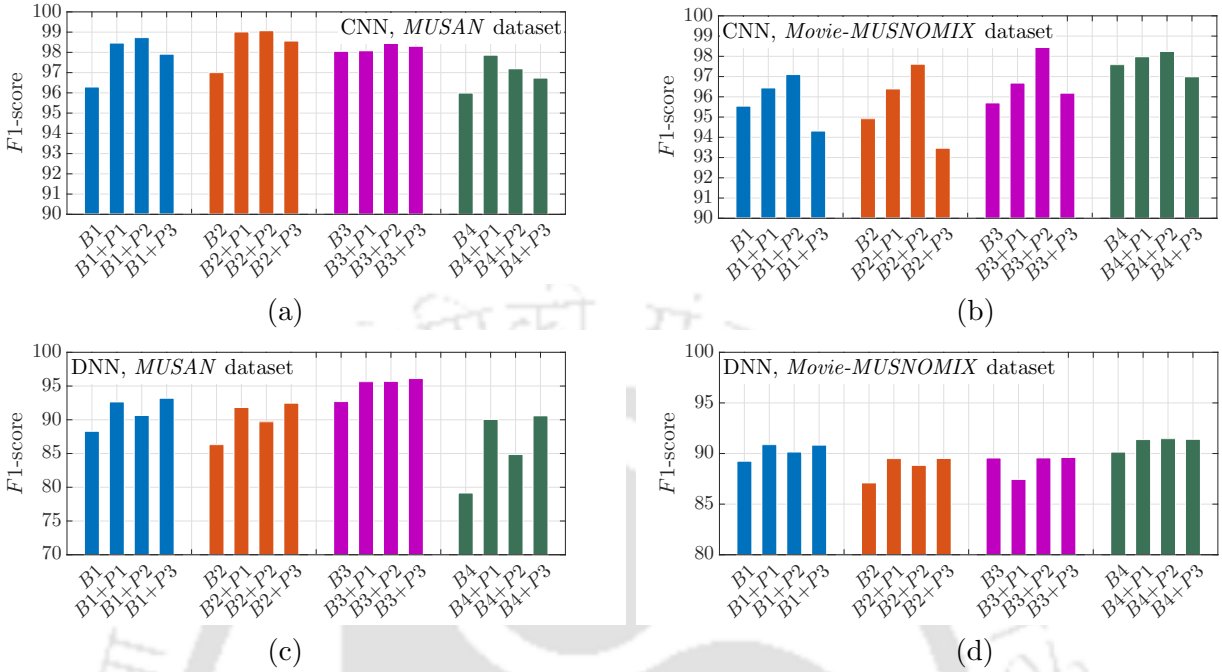


Figure 4.4: Illustrating the performance of the pair-wise combination of phase-based and magnitude-based features. Subfigures represent the performance of (a) CNN on *MUSAN* dataset, (b) CNN on *Movie-MUSNOMIX* dataset, (c) DNN on *MUSAN* dataset, and (d) DNN on *Movie-MUSNOMIX* dataset.

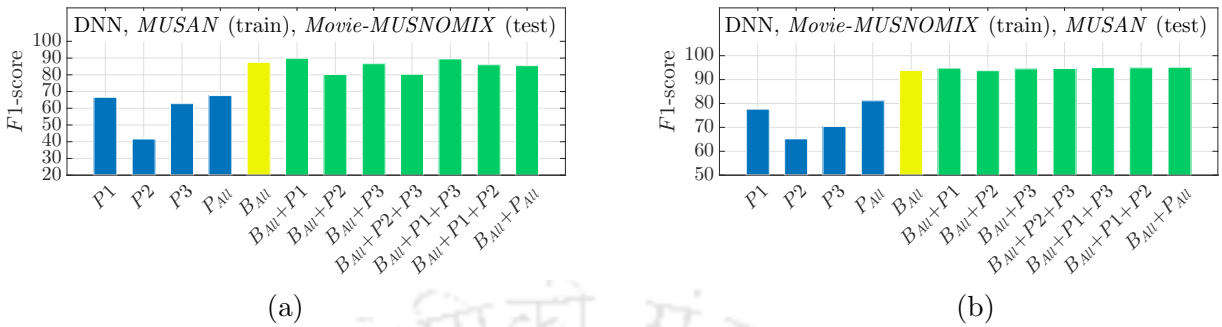
thus evident. However, a late fusion of the phase and magnitude-based features show significant improvement. The late-fusion combination is achieved by averaging the class-wise prediction scores of base classifiers trained on different features used in the ensemble. The class obtaining the highest combined score is selected as the predicted class. The feature $P2$ performs inferior to the other phase-based features for the *MUSAN* dataset. The feature $P3$ performs poorest for the *Movie-MUSNOMIX* dataset. These results follow the insights obtained from the statistical significance test performed using MANOVA (Fig. 4.3). The pair-wise combination of phase-based and magnitude-based features for both DNN and CNN classifiers over both the *MUSAN* and *Movie-MUSNOMIX* datasets provide an improvement over the baseline performance (see Fig. 4.4). Even the combination of all phase-based and magnitude-based features improves upon the combination of all magnitude-based features. However, only $\mathbf{B}_{All} + P3$ improves upon the performance of $\mathbf{B}_{All}$ for the CNN classifier trained over *Movie-MUSNOMIX* dataset. The lack of improvement with other combinations might be attributed to fewer training data in the Movie-MUSNOMIX dataset. The training data insufficiency problem may be solved by performing a transfer learning of the *MUSAN* models with *Movie-MUSNOMIX* data. It may be noted that the combinations of phase-based features with all the magnitude-based

Table 4.5: Class-wise performance analysis of phase-based and magnitude-based features for the *MUSAN* and *Movie-MUSNOMIX* datasets, using the CNN [2] classifier. Results are reported as "mean (μ) $F1$-score (over 3-fold cross-validation) $\pm$ standard deviation (σ)". The mean and standard deviation of $F1$-scores are expressed in percentage. Feature combinations that improve upon the baseline are highlighted with ▲.

			MUSAN		*Movie-MUSNOMIX*	
			Music	**Speech**	**Music**	**Speech**
		Features	$(\mu \pm \sigma)$	$(\mu \pm \sigma)$	$(\mu \pm \sigma)$	$(\mu \pm \sigma)$
Baseline	**Features**	$B1$	96.14 ±1.25	96.45 ±1.16	96.51 ±1.16	94.57 ±1.69
		$B2$	96.84 ±0.58	97.12 ±0.69	96.10 ±0.50	93.74 ±0.78
		$B3$	97.52 ±0.81	97.75 ±0.83	97.07 ±0.76	95.48 ±1.23
		$B4$	96.23 ±1.21	96.75 ±0.72	98.06 ±0.95	97.13 ±1.18
Phase	**Features**	$P1$	93.50 ±1.71	94.30 ±1.25	92.77 ±2.51	89.06 ±2.57
		$P2$	85.48 ±7.32	88.00 ±5.73	95.59 ±0.21	93.20 ±0.41
		$P3$	85.68 ±1.35	87.84 ±2.56	82.11 ±1.08	75.24 ±0.83
All	**Combinations**	$B1$+$B2$+$B3$+$B4$ ($\mathbf{B}_{All}$)	99.80 ±0.05	99.82 ±0.05	99.65 ±0.02	99.21 ±0.11
		$P1$+$P2$+$P3$ ($\mathbf{P}_{All}$)	95.67 ±1.25	96.13 ±1.26	96.88 ±0.47	93.06 ±1.27
		$\mathbf{B}_{All}$+$P1$	99.78 ±0.04	99.80 ±0.04	99.41 ±0.09	98.26 ±0.10
		$\mathbf{B}_{All}$+$P2$	99.82 ±0.08 ▲	99.84 ±0.07 ▲	99.48 ±0.05	98.83 ±0.33
		$\mathbf{B}_{All}$+$P3$	99.80 ±0.06	99.81 ±0.06	99.65 ±0.04	99.21 ±0.11
		$\mathbf{B}_{All}$+($P2$+$P3$)	99.87 ±0.04 ▲	99.88 ±0.04 ▲	96.88 ±0.03	93.06 ±0.16
		$\mathbf{B}_{All}$+($P1$+$P3$)	99.81 ±0.07 ▲	99.83 ±0.06 ▲	99.48 ±0.14	98.83 ±0.26
		$\mathbf{B}_{All}$+($P1$+$P2$)	99.84 ±0.02 ▲	99.85 ±0.02 ▲	99.48 ±0.11	98.45 ±0.10
		$\mathbf{B}_{All}$+$\mathbf{P}_{All}$	99.74 ±0.05	99.77 ±0.04	99.41 ±0.17	98.26 ±0.25

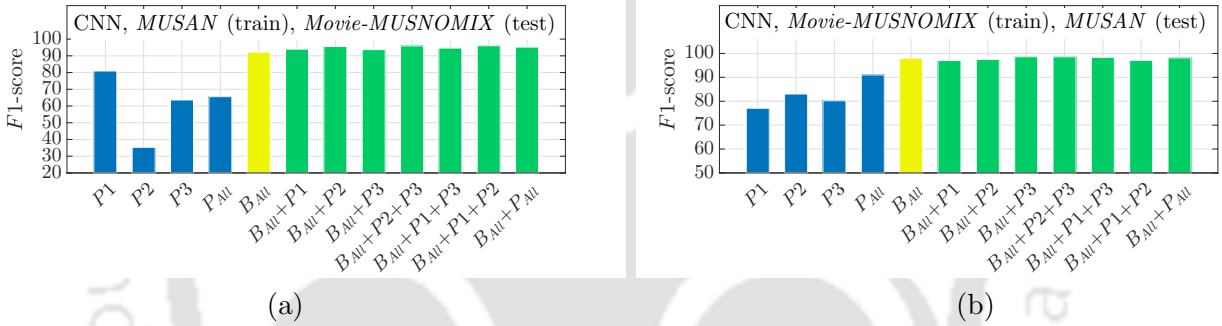
features ($\mathbf{B}_{All}$) do not provide very significant improvements. Whereas the pair-wise combinations of phase-based and magnitude features generally provide appreciable improvements. A possible reason for such a phenomenon might be that the $\mathbf{B}_{All}$ combination already provides high performances on its own. Therefore, the scope for improvement in combination with $\mathbf{B}_{All}$ might be significantly less.

In Table 4.5, the class-wise performances of the CNN classifier on the *MUSAN* and *Movie-MUSNOMIX* dataset are listed. It can be observed that the pair-wise combinations of phase-based and magnitude-based features improve the performance of both classes in most cases. For the *MUSAN* dataset, the performance improvement of the speech class tends to be more than the music class. Such a variation might be attributed to the efficacy of phase-based features. The performance of phase



(a) (b)

Figure 4.5: Illustrating the generalization performance of the **DNN** classifier for all the feature sets and their combinations, (a) trained on *MUSAN* dataset and tested on *Movie-MUSNOMIX*, (b) trained on *Movie-MUSNOMIX* dataset and tested on *MUSAN*. The mean $F1$-score of 3 folds (in %) is shown in bar-plots. Some combinations of the phase-based features with baseline features provide improvements ($B_{All} + P1$).



(a) (b)

Figure 4.6: Illustrating the generalization performance of the **CNN** classifier for all the feature sets and their combinations, (a) trained on *MUSAN* dataset and tested on *Movie-MUSNOMIX*, (b) trained on *Movie-MUSNOMIX* dataset and tested on *MUSAN*. The mean $F1$-score of 3 folds (in %) is shown in bar-plots. Some combinations of the phase-based features with baseline features provide improvements ($B_{All} + P_{All}$).

feature based speech detection is observed to be better than that of music for the *MUSAN* dataset. For the *Movie-MUSNOMIX* dataset, no class-specific trend is observed. The $P1$ and $P2$ features are known to capture the formant information in speech [243, 244]. Thus, the phase-based features may perform better in the case of speech detection compared to music.

4.4.3 Generalization performance

Generalization performance of the individual phase and magnitude-based features and their combinations is provided in Fig. 4.5 for the DNN classifier. The generalization performance for the CNN classifier is illustrated in Fig. 4.6. Results are presented in the form of bar charts. The performance trend obtained is similar to that observed with test signals from the same dataset (Table 4.4). Among the phase-based features, $P2$ exhibits the most inferior performance in all cases except when CNN trained on *Movie-MUSNOMIX* is tested on the *MUSAN* dataset. Some pair-wise combinations of

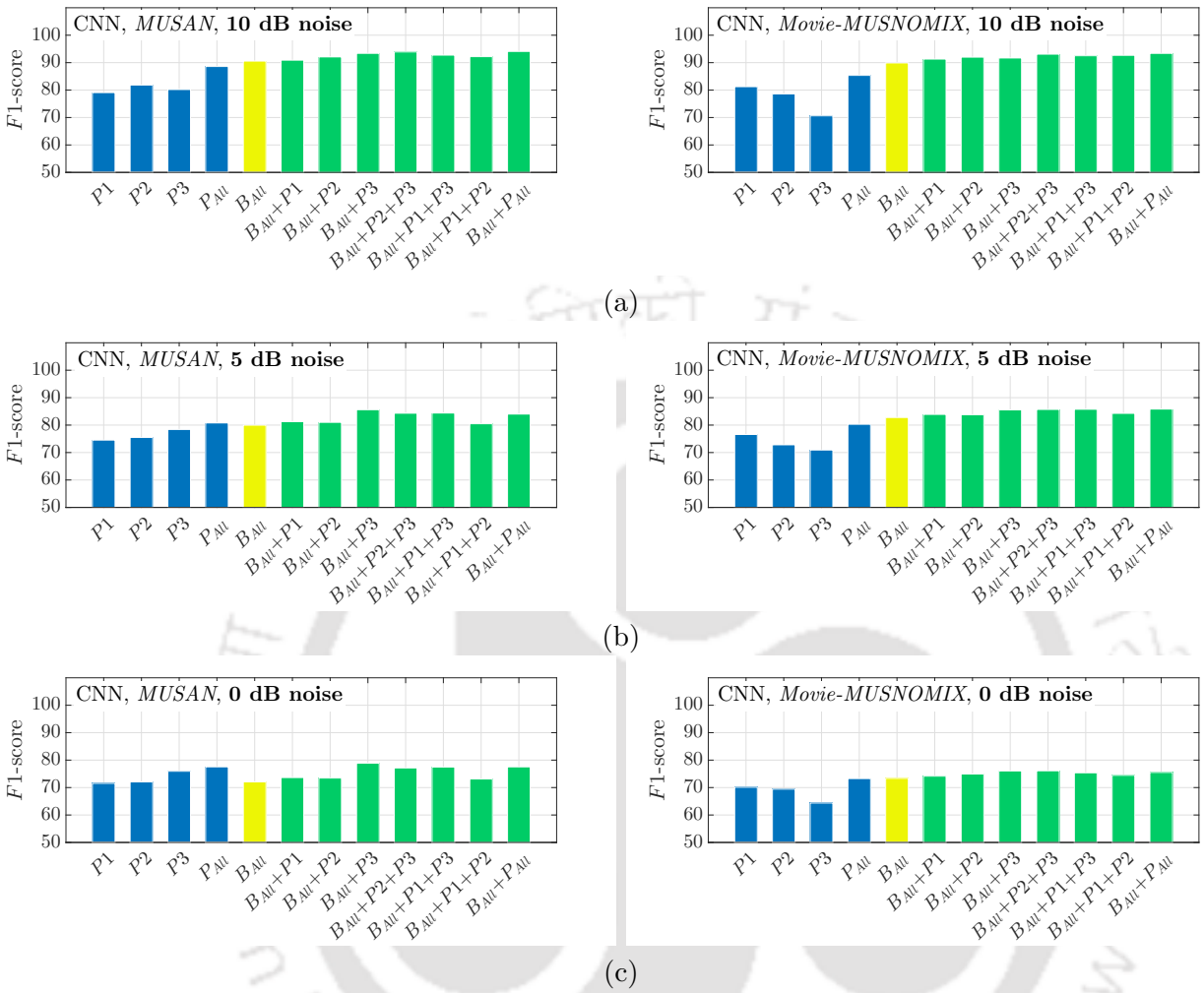


Figure 4.7: Illustrating the results of performing SMC using CNN classifier in different SNR conditions using various feature sets and their combinations on the *MUSAN* and *Movie-MUSNOMIX* datasets. The subfigures (a), (c) and (e) indicate the results computed for 10 dB noise, 5 dB noise, and 0 dB noise on the *MUSAN* dataset, respectively. The subfigures (b), (d), and (f) indicate the results computed for 10 dB noise, 5 dB noise, and 0 dB noise on the *Movie-MUSNOMIX* dataset, respectively.

phase and magnitude-based features still improve the baseline performance, while others fail. Such results indicate that few combinations of phase-based features with magnitude-based ones can assist even in the cross-dataset scenarios.

4.4.4 Effect of noise

The performances reported in Table 4.4 are computed using clean speech and music signals. Here, clean signal refers to the original recordings available with the datasets that are not corrupted with additional noise. However, the classification system will mostly encounter noisy signals in real-life scenarios. Thus, it becomes pertinent that any proposed SMC system is tested for robustness in the

presence of noise. In this context, a set of experiments are designed for this very purpose.

Noisy datasets are created by corrupting clean signals with noise signals at three different SNR levels, viz., 10dB, 5dB, and 0dB. The noise data are taken from the respective datasets. For the *MUSAN* dataset, the noise samples available with the dataset are used. Similarly, the corrupted signals for the *Movie-MUSNOMIX* dataset are generated by adding samples available in the noise class of this dataset. In most practical cases, speech and music signals might be corrupted by noise. A randomly selected speech or music signal from a clean signal dataset is mixed with a randomly selected noise signal at a particular SNR level. Similarly, three noisy versions (corresponding to three SNR levels) of *MUSAN* and *Movie-MUSNOMIX* datasets have been generated. The CNN models learned on clean data (used to generate the results reported in Table 4.4) are tested on the noisy signals.

As observed in Fig. 4.7, the presence of noise substantially affects the performance of both phase-based and baseline features. All feature sets perform best in the least noisy 10dB SNR case. Performance of all features gradually drops as the noise level increases to 0dB SNR. Among the baseline features, $B4$ seems to be the worst affected by the presence of noise. Since $B4$ is simple Mel-filter-bank energies, the presence of noise appears to affect it dramatically. Other features that employ DCT compaction (like $B3$, $P1$, and $P2$) tend to fare well in the noisy scenario. It is interesting to note that the combination of features (phase and magnitude) improves the performance of magnitude features. Improvement in performance provided by feature combinations seems better for MUSAN, suggesting that better models are trained when larger datasets are available. Thus, phase-based features also show their effectiveness in the presence of noise.

4.4.5 Audio segmentation

Closed-set classification tasks are comparatively easier than open-set segmentation problems. Hence, an additional set of experiments for segmenting continuous speech and music signal sequences is designed to establish phase information's effectiveness in the current context. The MIREX Speech/Music detection challenge 2018 [258] evaluation strategies have been followed in the segmentation experiments. Performance measures reported for this task are the same as those used in the MIREX challenge. In the MIREX challenge, two types of performances are reported. First, results reported at the level of short-term frames are termed segment-level performance. Second, true transition point detection results within acceptable tolerance windows are reported and termed event-level performance. The MIREX baseline results have been directly quoted from their website [258]. The proposed

Table 4.6: Event-level performance on synthetically concatenated speech and music files from *Movie-MUSNOMIX* and *MUSAN* datasets [203]. The performance measures for detecting both onset and offset points at different tolerance durations are reported.

	Tolerance (in ms)	Feature	Precision	Recall	F-measure	Deletion Rate	Insertion Rate	Error Rate
Movie MUSNOMIX	1000	$\mathbf{B}_{All}$	0.7058 ±0.04	0.9704 ±0.03	0.8165 ±0.03	0.0296 ±0.03	0.4087 ±0.09	0.4383 ±0.09
		$\mathbf{B}_{All}$ + $P1+P2$	0.7188 ±0.05	0.9902 ±0.02	**0.8319 ±0.03**	0.0098 ±0.02	0.3932 ±0.11	0.4030 ±0.10
	500	$\mathbf{B}_{All}$	0.6987 ±0.05	0.9604 ±0.03	0.8082 ±0.04	0.0396 ±0.03	0.4187 ±0.10	0.4582 ±0.10
		$\mathbf{B}_{All}$ + $P1+P2$	0.7111 ±0.05	0.9804 ±0.03	**0.8233 ±0.03**	0.0196 ±0.03	0.4030 ±0.10	0.4227 ±0.08
MUSAN	1000	$\mathbf{B}_{All}$	0.3044 ±0.11	1 ±0.00	0.4597 ±0.13	0 ±0.00	2.6107 ±1.41	2.6107 ±1.41
		$\mathbf{B}_{All}$ + $P1+P2$	0.4528 ±0.12	1 ±0.00	**0.6171 ±0.12**	0 ±0.00	1.3290 ±0.70	1.3290 ±0.70
	500	$\mathbf{B}_{All}$	0.3023 ±0.12	0.9930 ±0.00	0.4564 ±0.13	0.0070 ±0.00	2.6178 ±1.41	2.6248 ±1.41
		$\mathbf{B}_{All}$ + $P1+P2$	0.4528 ±0.12	1 ±0.00	**0.6171 ±0.12**	0 ±0.00	1.3290 ±0.70	1.3290 ±0.70

method's speech/music segmentation results are reported using two approaches. First, the event-level performance for synthetic concatenation of speech and music signals from the *Movie-MUSNOMIX* and *MUSAN* datasets are reported (Table 4.6). This approach is chosen to test how the trained CNN classifiers perform in the most basic setup without additional training for audio segmentation. The segment-level performance on these datasets is not reported separately since those results will be similar to the classification results reported in Table 4.4. Second, event-level (Table 4.9) and segment-level (Table 4.7 and Table 4.8) results are reported for real continuous audio segments of speech and music signals obtained from the *Muspeak* [205] dataset. CNN classifiers trained on the MUSAN dataset were used to obtain frame-level (speech or music) probabilities for the signals from the *Muspeak* dataset. Further, a Viterbi decoding based segmentation was performed by training a two-state Hidden Markov Model (HMM) on the frame-level probabilities obtained for the training signals from the *Muspeak* dataset. Finally, speech and music detection performance of phase-based features are also reported for the *DAFx-12* [206] (Table 4.10)

In Table 4.6, the segmentation performance on synthetically concatenated speech and music signals

Table 4.7: Segment-level performance of **music detection** performed on *Muspeak* dataset [205]. $\mathbf{B}_{All}$ indicates the group of baseline magnitude-based features, and $\mathbf{B}_{All} + \mathbf{P}_{All}$ indicates the combination of magnitude-based and phase-based features used in this chapter.

Method	Dataset	Acc	Music			Non-Music		
			Prec	Rec	F1-score	Prec	Rec	F1-score
Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.6860	0.905	0.3873	0.5424	0.6294	0.9624	0.7611
Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.9257	0.9751	0.8950	0.9334	0.8694	0.9683	0.9162
$\mathbf{B}_{All}$	Muspeak [205]	0.8841	0.9289	0.9127	0.9181	0.7330	0.6062	0.6345
$\mathbf{B}_{All} + \mathbf{P}_{All}$	Muspeak [205]	0.8704	0.9001	0.9378	0.9136	0.8539	0.4096	0.4221

are reported for different tolerance windows (500ms and 1000ms). The results are reported as $\mu \pm \sigma$ over 3-fold cross-validation. The CNN classifiers are trained using files from two folds for every iteration, while the third fold is kept aside for testing. The experiment is performed by first concatenating signals from two random files from speech and music classes of the test fold. Then, the trained CNN classifier predicts the frame-level class-wise probabilities for frames in the concatenated signal. The frame-level probabilities are then smoothed using a median filter. Finally, the smoothed probabilities are thresholded to obtain class labels. The thresholded class labels are first-order differenced to generate the segment transition points. Class-specific results are not reported since no separate sequence modeling is done for this experiment. This experiment is performed for both the *Movie-MUSNOMIX* and *MUSAN* datasets. As can be observed from Table 4.6, the segmentation results are quite good. Such a performance can be expected for the synthetic concatenation of signals. However, the important point to be noted here is that $\mathbf{B}_{All} + P1 + P2$ performs better than $\mathbf{B}_{All}$ for all the tolerance windows considered. Such a result indicates that a combination of phase-based and magnitude-based features can improve the segmentation performance.

The second segmentation experiment is performed over the *Muspeak* dataset [205] which contains signals with continuous transitions of speech and music. For this experiment, the following procedure is followed to perform the segmentation. First, the signals from *Muspeak* are divided into 3 folds. Then the CNN classifiers trained on the MUSAN dataset are used to obtain frame-level class probabilities

Table 4.8: Segment-level performance of **speech detection** evaluated on *Muspeak* dataset [205]. $\mathbf{B}_{All}$ indicates the group of baseline magnitude-based features and $\mathbf{B}_{All}$+$\mathbf{P}_{All}$ indicates the combination of magnitude-based and phase-based features used in this chapter.

Method	Dataset	Acc	Speech			Non-Speech		
			Prec	Rec	F1-score	Prec	Rec	F1-score
Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.8770	0.9090	0.9285	0.9186	0.7751	0.7251	0.7493
Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.9617	0.9603	0.9564	0.9583	0.9633	0.9662	0.9648
$\mathbf{B}_{All}$	Muspeak [205]	0.8888	0.7690	0.5931	0.6442	0.9180	0.9321	0.9235
$\mathbf{B}_{All}$ + $\mathbf{P}_{All}$	Muspeak [205]	0.8742	0.5988	0.4000	0.4249	0.8899	0.9574	0.9189

for the signals in each fold of the *Muspeak* dataset. These frame-level probabilities are smoothed using a median filter to remove outliers. A two-state left-to-right Hidden Markov Model (HMM) is trained on the smoothed frame-level probabilities. Each state of the HMM is modeled as a 2 mixture GMM.

One HMM state is initialized with the parameters of a bimodal GMM trained on frame-level probabilities of speech signals. Another bimodal GMM trained on frame-level probabilities of music signals is used to initialize the other HMM state. The starting state and state-transition probabilities of the HMM are randomly initialized. During the training of the HMM, parameters of the state-emission distributions are not updated. Only the starting state and the state transition probabilities are learned from training data sequences using the Expectation-Maximization algorithm. After training an HMM for a particular training fold, it is used to generate a state sequence for the signals in the corresponding test fold of the *Muspeak* dataset. The obtained state sequence is first-order differenced to obtain the transition points.

Segment-level performance results reported in Table 4.7 and Table 4.8 indicate that the proposed method performs comparitively poorer than the best submission to the MIREX 2018 challenge. The results reported for the MIREX baseline are computed on unreleased evaluation datasets different from those used in this chapter. Hence, a direct comparison of results is not possible. Moreover, a limited number of training datasets are used in this work compared to multiple training datasets used in MIREX 2018 challenge. Hence, the performances obtained in this experiment are relatively low for

the proposed methods.

The event-level performance of detecting onset and offset points of speech and music segments are reported in Table 4.9. The segmentation boundaries are detected mostly beyond the limits of tolerance windows by the proposed method. Such a lack of precision results in poor performance. The performance improves with a larger tolerance window of 5000ms. Also, many spurious boundaries have been detected that increase the number of false positives. Such poor results might be due to a couple of reasons. First, the CNN classifiers trained on the *MUSAN* dataset generate the frame-level probabilities of signals from the *Muspeak* dataset. The performance of classifiers trained on one dataset and tested on another is generally low. Second, the frame-level CNN classifiers are trained on comparatively clean speech and music signals from the *MUSAN* dataset. However, the *Muspeak* dataset contains speech-to-music transitions and overlapping regions. Hence, the CNN classifiers perform poorly in the transition and overlapping regions. This poor performance causes the detection of segment boundaries beyond tolerance ranges. Nevertheless, the performance trend for phase-based and magnitude-based feature combinations is interesting. It can be observed from Table 4.9 that the segmentation performance of $\mathbf{B}_{All} + \mathbf{P}_{All}$ is better than that of $\mathbf{B}_{All}$ for mostly all tolerance windows. The performance of $\mathbf{B}_{All}$ is slightly better than $\mathbf{B}_{All} + \mathbf{P}_{All}$ only for speech with a 5000ms tolerance window and for music with a 500ms tolerance window. For all other cases, $\mathbf{B}_{All} + \mathbf{P}_{All}$ performs better than $\mathbf{B}_{All}$. Such results justify that the combination of phase-based features with magnitude-based features can improve speech-music segmentation performance.

Table 4.10 lists the results obtained with phase-based features on the *DAFx-12* dataset. Performance of all baseline features combination (B_{All}) and the best three performers among the combinations of phase and magnitude features on both the test sets from *DAFx-12* dataset are reported. The best performers are selected separately for speech and music detection. The baseline results for the *DAFx-12* dataset are better than that of the phase-based feature combinations. However, the proposed feature combinations provide comparable precision for music detection on both test sets. Comparable recall values are also obtained with the proposed feature combination for speech detection on the Austrian test set. Similar results were obtained in chapter 3. The lack of performance improvements may therefore be attributed to the presence of extreme data imbalance and speech overlapped with music signals in the *DAFx-12* dataset. These challenges are not explicitly tackled in this chapter's proposal. Hence, the obtained performances are observed to be significantly poor. Nevertheless, the

Table 4.9: Event-level performance on *Muspeak* dataset [205]. The onset-offset $F1$-score at different tolerance durations is reported. $\mathbf{B}_{All}$ indicates the group of baseline magnitude-based features and $\mathbf{B}_{All} + \mathbf{P}_{All}$ indicates the combination of magnitude-based and phase-based features used in this chapter. A discussion on the performance is provided in subsection 4.4.5.

		Dataset	**F1-score** (500ms)	**F1-score** (1000ms)	**F1-score** (5000ms)
Music	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.0930	0.1142	–
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.2235	0.2480	–
	$\mathbf{B}_{All}$	Muspeak [205]	0.0214 ±0.02	0.0550 ±0.02	0.2184 ±0.16
	$\mathbf{B}_{All} + \mathbf{P}_{All}$	Muspeak [205]	0.0177 ±0.02	0.0677 ±0.08	0.2235 ±0.23
Speech	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.1603	0.2122	–
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.4139	0.4350	–
	$\mathbf{B}_{All}$	Muspeak [205]	0.0126 ±0.01	0.0371 ±0.02	0.2869 ±0.16
	$\mathbf{B}_{All} + \mathbf{P}_{All}$	Muspeak [205]	0.0306 ±0.03	0.0678 ±0.07	0.2637 ±0.23

combination of phase-based features with magnitude-based ones provides better performance than the combination of baseline features (B_{All}) in all cases, except for speech detection in the Austrian test set. Thus, it may be claimed that phase information can aid in the speech-music detection task in combination with magnitude-based features.

4.4.6 Discussions

A direct comparison with some state-of-the-art results from literature over the datasets used in this chapter are provided in Table 4.11. First, Doukhan et al. [211] reported results for SMC using the MFCC features with a CNN classifier. Their performances on *GTZAN* and *Scheirer-Slaney* datasets are found to be better (on account of a better classifier) compared to the results of this proposal (using naive Bayes and SVM classifiers). However, the proposed combination of phase-based and magnitude-based features provides better performance over the *MUSAN* dataset. Second, Hussain et al. [259] reported their results on the *GTZAN* dataset using deep-learning-based classifiers. Naturally, their performance is better than the SVM classifier used for the *GTZAN* dataset in this chapter. Third, the proposed method performs better than the results reported by Li et al. [260] on the *MUSAN* dataset. Fourth, the results reported by Birajdar et al. [261] are very high for all the datasets. However, their

Table 4.10: Performance of B_{All} and the top 3 combinations of phase-based and magnitude-based features on **real signals** from the *DAFx12-dataset* [206] are tabulated here. Baseline results are quoted directly from the reference.

		Music/ Non-music				**Speech/ Non-speech**			
Test set	**Method**	**Acc**	**Prec**	**Rec**	**F1**	**Acc**	**Prec**	**Rec**	**F1**
Swiss	Schlüter et al. [206]	97.30	98.80	98.00	98.40	98.40	96.40	96.50	96.40
	B_{All}	87.47	98.16	86.86	92.16	88.03	68.17	91.58	78.16
	$P2$+$B1$	90.46	95.14	93.53	**94.33**	–	–	–	–
	$P2$+B_{All}	88.77	98.03	88.54	**93.04**	–	–	–	–
	$P1$+$B1$	88.08	96.9	88.8	**92.67**	–	–	–	–
	$P1$+P2+B_{All}	–	–	–	–	93.06	85.24	85.05	**85.15**
	$P1$+B_{All}	–	–	–	–	92.63	82.93	86.24	**84.55**
	P_{all}+B_{All}	–	–	–	–	92.51	82.75	85.87	**84.29**
Austrian	Schlüter et al. [206]	95.60	95.30	97.40	97.30	97.00	95.90	95.10	95.50
	B_{All}	84.92	97.17	84.16	90.20	92.29	87.92	90.45	89.17
	$P2$+$P3$+B_{All}	86.29	95.91	87.08	**91.29**	–	–	–	–
	$P1$+$P3$+B_{All}	85.86	96.33	86.13	**90.94**	–	–	–	–
	$P3$+B_{All}	85.97	96.05	86.54	**91.05**	–	–	–	–
	$P2$+B_{All}	–	–	–	–	92.04	86.99	90.91	88.91
	$P3$+B_{All}	–	–	–	–	90.97	83.75	92.16	87.75
	$P2$+$P3$+B_{All}	–	–	–	–	90.73	83.01	92.52	87.51

results are not directly comparable to the proposed work. Birajdar et al. [261] used only subsets of the datasets to generate results while this work considers the whole datasets. Fifth, performances reported by Khonglah et al. [148] and Sell et al. [140] are computed using a context window size of 1s. Thus, their results are better than that reported in this chapter which uses a smaller context window size ($\approx$ 695ms).

The performances of phase-based features (Table 4.4) show that the phase carries crucial information that can be used to discriminate between speech and music. There are three primary takeaways from this chapter. First, HNGDCC ($P1$) can be considered a decent feature for SMC. The successful performance of $P1$ may be attributed to the fact that the HNGD spectrum enhances the formant peaks [243], and music signals do not have any formant structure. Second, the IFCC feature ($P3$) represents the phase differences between the underlying signal's NB components. IFCC can be con-

Table 4.11: Comparison of state-of-the-art speech/music classification with the performances obtained in this chapter. The state-of-the-art results are quoted directly from the literature.

Paper	Feature	Classifier	Metric	Dataset		
				GTZAN	*Scheirer-Slaney*	*MUSAN*
Doukhan et al. [211]	MFCC	CNN	Recall	97.21	92.79	99.11
Hussain et al. [259]	MFCC	SwishNet	$F1$-score	98.17	–	–
Hussain et al. [259]	Log Melspectrogram	MobileNet	$F1$-score	98.77	–	–
Li et al. [260]	12-dim statistical feature	DNN	Accuracy	–	–	94.08
Birajdar et al. [261]	Chromagram visual feature	SVM	Accuracy	100	100	100
Birajdar et al. [261]	Chromagram spectral feature	SVM	Accuracy	95	95	98.48
Khonglah et al. [148]	Speech-specific features & others	SVM	Accuracy	92.03	96.75	–
Sell et al. [140]	Chroma high frequency & others	GMM	Accuracy	93.50	–	–
Proposed	$\mathbf{B}_{All} + \mathbf{P}_{All}$	SVM	$F1$-score	91.32 $\pm$2.60	–	–
Proposed	$\mathbf{B}_{All}$+$P1$	SVM	$F1$-score	–	93.07 $\pm$1.22	–
Proposed	$\mathbf{B}_{All}$+$P2$+$P3$	CNN [2]	$F1$-score	–	–	99.87 $\pm$0.03

sidered another useful feature in the current task. Third, the MGDCC feature ($P2$) does not perform well in every case. Previously, Murthy et al. [244] observed that $P2$ performed poorly in detecting unvoiced phonemes in a phoneme recognition task. In a similar vein, it can be said that $P2$ is affected by the presence of unvoiced segments in speech and tone onsets in music. It can now be summarised that the magnitude-based features perform better individually. Nevertheless, phase-based features provide competing performances in the current task. The phase-based features carry enough comple-

mentary information to improve the baseline performance with the magnitude-based features. Hence, a combination of phase and magnitude-based features should be used to enhance SMC's performance.

The issues encountered with segmenting real continuous audio segments might be tackled by training the frame-level CNN classifiers on speech-music transition data, overlapped speech-music data, and clean data. The segmentation problem is not explored further as this chapter aims to study the effectiveness of phase-based and magnitude-based features in speech vs. music classification. The trends obtained with the preliminary segmentation experiments validate the claim that the combination of phase-based and magnitude-based features can improve both classification and segmentation performance.

4.5 Summary

Existing works on SMC indicate a limited use of phase information in the task. The phase component of speech and music signals carry distinct properties. In this regard, three existing phase-based audio features, viz. HNGDCC, MGDCC, and IFCC are explored in this chapter. The phase-based features show a low correlation with magnitude-based features and thus indicate the presence of complementary information. Mostly, magnitude-based baseline features perform better than the phase-based ones individually. However, the performances of magnitude-based features are further enhanced when combined with the phase-based ones through late fusion. Thus, it becomes imperative that the phase information is used to build more robust SMC systems. The combination of phase and magnitude-based features shows stable cross-dataset performance. The phase and magnitude-based feature combinations perform comparatively well even when test signals are corrupted with mild to heavy noise. Last but not least, phase features in combination with magnitude-based features even aid in better segmentation of continuous speech-music sequences.

Till now, the focus of this thesis has been on the classification performance improvement of isolated speech and music signals. However, a significant portion of speech and music signals are found as overlapped mixtures in real scenarios. Detection of such overlapping signals is more challenging than detecting the non-overlapping signals. Hence, the next chapter 5 attempts to discriminate overlapped speech and music signals from isolated ones. In this regard, features obtained from harmonic-percussive source separation and classifiers based on the multi-task learning framework are explored.

5

Harmonic-Percussive Features for Speech Music Overlap Detection

Publications

- **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Clean vs. Overlapped Speech-Music Detection Using Harmonic-Percussive Features and Multi-Task Learning," in *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, DOI: 10.1109/TASLP.2022.3164199, 2022.

Contents

Objective

The previous two chapters focused on developing efficient features for the classification of isolated speech and music signals. However, in most practical cases (e.g., movie audio sequences), speech and music are frequently encountered in overlapped conditions. For example, movie editors add background music to scenes with dialogues to highlight a particular mood. Therefore, detecting speech and music signals in isolated and overlapped conditions is an essential pre-processing step for any movie audio processing application. Hence, this chapter aims to propose features and classification frameworks for efficient detection of speech and music signals in overlap or otherwise. Speech signals have wavy and continuous harmonics, while music signals exhibit horizontally linear and discontinuous harmonic patterns. Also, the music signals contain more percussive components than speech signals. These percussive components are manifested as vertical striations in the spectrograms. In the case of speech music overlap, it might be challenging for automatic feature learning systems to extract class-specific horizontal and vertical striations from the combined spectrogram representation. We believe a pre-processing step that separates the harmonic and percussive components of the spectrograms might help the classifier learn discriminative representations. Accordingly, this chapter proposes the use of the harmonic-percussive source separation method for the generation of appropriate features to detect speech overlapping with music. Additionally, this chapter also explores the traditional and cascaded-information Multi-Task Learning (MTL) frameworks to design better classifiers. The MTL framework aids the training of the main task by employing simultaneous learning of several related auxiliary tasks. The experimental results are reported on synthetically generated overlapped speech music signals and natural recordings. Four state-of-the-art approaches are used for performance comparison. Experiments show that harmonic and percussive decompositions of spectrograms perform better as features. Moreover, classifiers with the MTL framework further improve performance.

5.1 Task overview

Speech and music are the most frequently encountered audio categories in movies, TV shows, web series, and radio broadcasts. Speech vs. Music Classification (SMC) is a well-researched problem. State-of-the-art methods [2, 212, 262, 263] can identify isolated speech and music segments with impressive accuracy. However, speech and music are often found in overlapped conditions in most practical scenarios. For example, sentimental scenes in movies and TV shows frequently have speech

with background music to highlight the scene's mood. If such segments are not identified beforehand and processed separately, these may disrupt the performance of high-level applications like automatic speech recognition (ASR) and music information retrieval. Hence, this chapter discriminates isolated speech and music segments from their overlapping mixtures.

Initial studies in speech overlapped with music detection were performed using traditional feature engineering approaches and machine learning algorithms. Some authors dealt with the presence of background music by compensating for it [264], suppressing it using Non-Negative Matrix factorization [133] or separating it using Independent Component Analysis [265]. Others detected the presence of background music using autocorrelation-based features [123] or Principal Component Analysis [124]. Some works attempted to segment overlapping soundtracks by using Singular Spectrum Analysis [153] or Self-Similarity Matrix-based approach [266]. Most works used classifiers like Gaussian Mixture Models (GMM) [123, 264], Support Vector Machines (SVM) [123, 264, 266], Random Forests [266] and Logistic Regression [266].

Deep-learning-based algorithms have also been explored in the task of speech overlapped with music detection. Jia et al. [155] detected the presence of music using a novel Hierarchical Regulated Iterative Network. In contrast, Gimeno et al. [154] used Recurrent Neural Network trained on limited data for the task. Venkatesh et al. [267] used Convolutional Recurrent Neural Network with artificially synthesized radio-broadcast like speech and music data. Bhattacharjee et al. [268] used enhanced spectrograms called pyknograms for the task of speech overlapped with low-energy music detection with a fully convolutional network.

In this context, it is relevant to review the popular Albayzín campaigns which are a set of audio processing challenges open for public participation. Audio Segmentation Evaluation (ASE) was one of the tasks in their 2010, 2012 [269] and 2014 [270] editions. In the Albayzín-2014 ASE, participating systems were required to identify the presence of speech, music, or noise, either isolated or overlapped. Albayzín-2014 ASE provided a more general and realistic database than those used in the Albayzín-2010 and 2012 ASE. The submitted systems used two distinct approaches for the task [270]. About half of the submissions followed the segmentation-and-classification strategy using techniques like Bayesian Information Criterion. The remaining systems employed a segmentation-by-classification strategy whereby models of individual classes are used to generate smoothed predictions in a post-processing step. The GMMs followed by Hidden Markov Models (HMM) was a popular choice for this strategy.

The most common feature choices were Mel Frequency Cepstral Coefficients (MFCC), Perceptual Linear Prediction coefficients, short-term energy, and other standard spectral features. GMMs, i-vectors, HMM, Logistic Regression, and SVM were popularly used for classification. The participating authors observed that the presence of noise class increased the task's difficulty. Overlapping speech, music, and noise segments were mostly confused with respective pair-wise overlaps. Many recent works have also tried to solve the Albayzín-2014 ASE task using deep-learning-based approaches. Gimeno et al. [271] employed a sequence of Bi-directional Long Short Term Memory units to segment audio sequences, followed by classification.

Another popular challenge, known as the Music Information Retrieval Evaluation eXchange (MIREX), tasked the participants to detect speech and music in its 2015 and 2018 editions. Classification-and-segmentation and classification-by-segmentation were the primary approaches used in MIREX challenges as well. Frame energy, zero-crossing rates, spectral features, MFCCs, and Chroma-based features were popular features used in MIREX 2015. Most authors adopted Mel-spectrograms as the input feature in MIREX 2018. Classification systems based on heuristic-based decision functions, SVM, Restricted Boltzmann Machines, Logistic Regression, Random Forests, and single layer feedforward networks were popular in MIREX 2015. However, most systems switched to a deep-learning-based classifier in MIREX 2018.

Previous works in speech and music detection have used Mel-scaled spectrogram (MS) or its derivatives as the principal feature [267, 271, 272]. Few submissions to the MIREX challenges have explored Constant-Q Transform spectrograms and Periodograms. Other features used were Self-Similarity Matrix [266], Spectral Tracking [126], Continuous Frequency Activations [122], Pyknograms [268], MFCCs [141] and standard tempo-spectral features. To the best of our knowledge, all previous works have used a combined harmonic and percussive representation. Speech and music signals have distinct harmonic and percussive characteristics. Fig. 5.1(a)-(c) show the spectrograms of music, speech, and speech overlapped with music (speech+music) at 0dB Speech-to-Music Ratio (SMR), respectively. The harmonics in speech have a wavy structure, while music harmonics are relatively more stable (horizontally linear). Percussive components characterized by an impulse like vertical striations are found more in music [219] than in speech. It might be challenging for an automatic feature learning system to isolate and learn the class-specific patterns from a combined representation like a spectrogram. We believe that a separate presentation of the harmonic and percussive information might improve

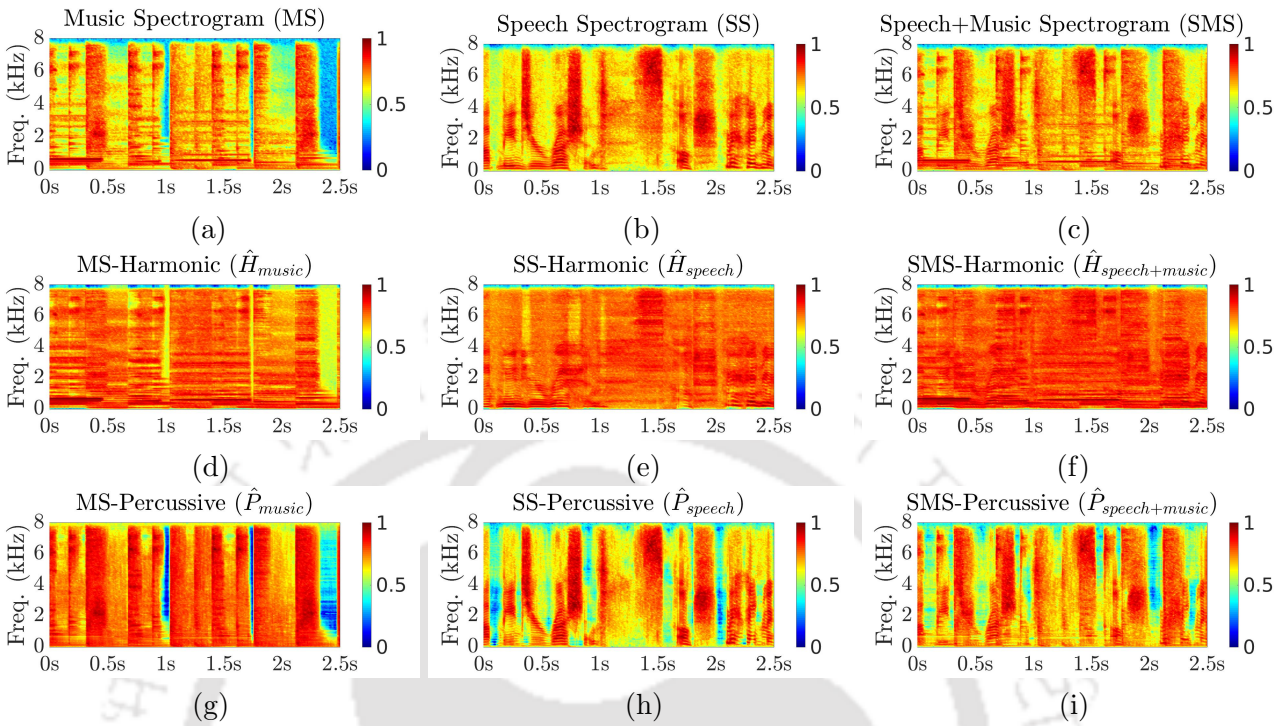


Figure 5.1: This figure illustrates the spectrograms of (a) music, (b) speech, and (c) speech+music mixed at 0dB, along with their harmonic (second row), and percussive (third row) decompositions. It may be noted that speech+music spectrograms carry the combined striations of both the component signals.

discriminative learning. This idea is the main motivation for using Harmonic-Percussive Source Separation (HPSS) to compute previously unexplored features in this task. The HPSS based features have been successfully used earlier in Jazz solo instrument classification [273], time-scale modifications of music signals [274], and music genre classification [275]. We believe that HPSS representations might perform better in the current task as well.

Another contribution of this chapter is the exploration of the Multi-Task Learning (MTL) framework in the context of the current task. The motivation for using MTL in the current task can be justified based on the following reasons. First, MTL has been successfully explored previously in different speech and audio processing applications with considerable success. Notable examples include speech recognition [276], speaker verification [277], harmony recognition of symbolic music [278], analysis of acoustic scenes and events [279], speech synthesis [280], end-to-end speech translation [281], neural machine translation [282] and several others. Second, the current task has some auxiliary information like mixing SMR ratio that can be used for additional supervision in training the classifier. Third, a model trained using this framework learns to perform multiple tasks for any given input.

For memory-constrained systems [283], such a model will be extremely beneficial. Fourth, training networks with highly related auxiliary tasks and sufficient noise levels inherent in the data can improve generalization capabilities [284]. Thus, the MTL framework may be regarded as a promising avenue of experimentation in the current task. In addition to the traditional MTL framework, this chapter also explores cascaded information in the MTL framework that is found to be beneficial in literature [285–287]. This chapter has four principal contributions.

(i) Exploration of HPSS based features for speech+music detection. In this context, various feature fusion strategies, viz. Early-Fusion (EF), Intermediate-Fusion (IF), and Late-Fusion (LF) are also investigated to identify the best feature combinations for the underlying task (see subsection 5.3.7).

(ii) Explorations of traditional and cascaded-information MTL frameworks for enhancement of classification performances.

(iii) Experimental analysis of challenging mixing SMRs (say -5dB and 20dB) [124, 133, 153, 264] on speech+music detection performance.

(iv) Experimental study of the proposed system on real signals containing isolated or overlapped speech and music.

The rest of the chapter is organized in the following manner. section 5.2 discusses the proposed approach for the detection of speech+music. A brief description of the procedure for HPSS is provided in subsection 5.2.1. An analysis of class separability provided by the proposed features is provided in subsection 5.2.2. The proposed MTL design is explained in subsection 5.2.3. The experiments performed and results obtained are discussed in section 5.3. Finally, the chapter is summarized in section 5.4.

5.2 Proposed feature and network architectures

This chapter explores representations obtained from HPSS as features to detect speech+music signals. Moreover, a classifier designed in the MTL framework to leverage additional implicit information associated with the underlying task is proposed. The following subsections describe the methodology for HPSS decomposition and the design of the proposed models in the MTL framework.

5.2.1 Harmonic-percussive source separation

This chapter uses the HPSS decomposition method proposed by Fitzgerald et al. [288]. Let, $\mathbf{X}$ be a complex-valued Discrete Fourier Transform (DFT) based spectrogram, and $|\mathbf{X}|$ be the magnitude spectrogram derived from $\mathbf{X}$. The spectrogram $|\mathbf{X}|$ can be further decomposed into separate harmonic and percussive components. A harmonic enhanced spectrogram ($\mathbf{X}_H$) is computed by median filtering the rows of $|\mathbf{X}|$ with a window size of l_{harm}. Similarly, a percussion enhanced spectrogram ($\mathbf{X}_P$) is computed by median filtering the columns of $|\mathbf{X}|$ with a window size of l_{perc}. The respective equations 5.1 and 5.2 are used for computing $\mathbf{X}_H$ and $\mathbf{X}_P$.

$$\mathbf{X}_H[k, 0:(L-1)] = \text{median}\left(|\mathbf{X}|\,[k, 0:(L-1)], l_{\text{harm}} \right) \tag{5.1}$$

$$\mathbf{X}_P[0:(N_f-1), l] = \text{median}\left(|\mathbf{X}|\,[0:(N_f-1), l], l_{\text{perc}} \right) \tag{5.2}$$

where, $k = 0, \ldots (N_f - 1)$ are the indices of N_f frequency bins in $\mathbf{X}$, $l = 0, \ldots (L-1)$ are the indices of L frames in $\mathbf{X}$, and median $(\bullet)$ signifies the median filter. Masks are generated using these enhanced spectrograms that are multiplied with the original spectrogram $\mathbf{X}$ to obtain the respective decompositions. Two variants of these masks can be computed, hard masks and soft masks. This chapter uses soft masks for decomposition. The computation of soft masks $\mathbf{M}_H$ and $\mathbf{M}_P$ are based on Wiener filtering using equations 5.3 and 5.4, respectively.

$$\mathbf{M}_H[k, l] = \frac{\mathbf{X}_H^2[k, l]}{\left(\mathbf{X}_H^2[k, l] + \mathbf{X}_P^2[k, l]\right)} \tag{5.3}$$

$$\mathbf{M}_P[k, l] = \frac{\mathbf{X}_P^2[k, l]}{\left(\mathbf{X}_H^2[k, l] + \mathbf{X}_P^2[k, l]\right)} \tag{5.4}$$

This chapter considers power spectrograms. These masks can also be computed using the magnitude spectrograms. These masks are element-wise multiplied ($\bigotimes$) with the original complex spectrogram ($\mathbf{X}$) to generate the harmonic (eqn. 5.5) and percussive (eqn. 5.6) decompositions.

$$\hat{\mathbf{X}_H} = \mathbf{M}_H \bigotimes \mathbf{X} \tag{5.5}$$

$$\hat{\mathbf{X}_P} = \mathbf{M}_P \bigotimes \mathbf{X} \tag{5.6}$$

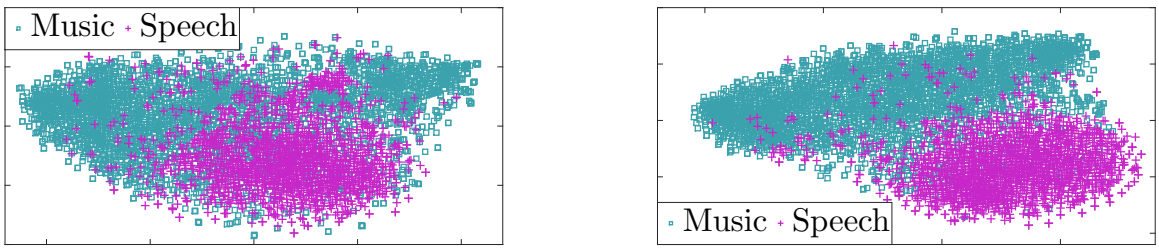
For a detailed treatment of the method, the reader is encouraged to refer to [288]. The Fig. 5.1(d)-(f) illustrate the harmonic decomposition ($\hat{\mathbf{X}}_H$) and Fig. 5.1(g)-(i) show the percussive decomposition ($\hat{\mathbf{X}}_P$) of the spectrograms in Fig. 5.1(a)-(c). It can be observed from the figures that $\hat{\mathbf{X}}_H$ can clearly capture the harmonic striations of the signal, while $\hat{\mathbf{X}}_P$ contains the signal's percussion patterns.

5.2.2 Class-separability provided by HPSS

This subsection describes a method employed to gauge the enhancement in class separability induced by HPSS. The linear harmonics in music might span only a few adjacent rows (along the frequency dimension) in the spectrogram. In contrast, harmonics in speech may span over many rows in the spectrogram because of their wavy nature. Therefore, the harmonic rows might have a localized energy distribution over successive audio frames in music but not in speech. It may be noted that the rows without any harmonics in either signal's spectrogram would mainly contain background information and not provide much separability. Alternatively, in the case of spectrogram columns containing percussive striations, the frame's energy is almost evenly distributed across all frequency bins. For non-percussive frames, the frame energy is contained only in a few frequency bins. Thus, energy distribution might be localized in spectrogram columns containing percussion and widely distributed otherwise. Music is believed to have a lot of percussive components [219], unlike speech. Hence, it may be expected that the respective row-energy and column-energy distributions of speech and music signals would be different.

The nature of row-energy and column-energy distributions is studied in this chapter by computing their skewness measures. Such distributions may be expected to have varying degrees of asymmetries. Skewness is chosen as it measures the asymmetry of a distribution. An increased class separability displayed by the skewness values of row and column-energy distributions might prove the effectiveness of HPSS decomposition in the current task.

This chapter uses spectrogram patches of $n_t = 68$ consecutive frames (695ms) for classification. Since the harmonics tend to span across multiple rows for both speech and music, the row-energy distributions of raw spectrograms might be noisy. Hence, the spectrograms are smoothed along the frequency axis using 21 Mel-scale filters. The choice of 21 Mel-filters is inspired by the work of Doukhan et al. [2] who obtained excellent performances with this setup. Thus, patches of Mel Spectrogram (MS), Mel Harmonic Spectrogram (MHS), and Mel Percussive Spectrogram (MPS) of size 21×68 are used to generate the class-separability visualizations. In this chapter, skewness is computed using



(a) (b)

Figure 5.2: The t-SNE plots illustrate the distribution of skewness vectors. The subfigures are generated by concatenating $\mathbf{r}_{skew}$ and $\mathbf{c}_{skew}$ vectors computed from $|S|$ (shown in (a)), and $\hat{\mathbf{X}_H}$ and $\hat{\mathbf{X}_P}$ (shown in (b)). It can be observed that harmonic and percussive decompositions can improve the class separability of speech and music. For more details, refer subsection 5.2.2.

the *SciPy* [289] Python library. The skewness value $s_R^{(i)}$ $(i = 0 \ldots 20)$ of each row in MS (or MHS) is computed to capture the class-specific harmonic information. These skewness values are concatenated to form the 21-dimensional vector $\mathbf{r}_{skew} = [s_R^{(0)}, \ldots s_R^{(20)}]$. The vector $\mathbf{r}_{skew}$ is used as a representation of the harmonic information in spectrograms of speech and music signals. Similarly, the skewness $s_C^{(j)}$ values computed from the columns of MS (or MPS) are concatenated to form the 68-dimensional vector $\mathbf{c}_{skew} = [s_C^{(0)}, \ldots s_C^{(67)}]$. The percussive information in spectrograms of speech and music are represented by their corresponding $\mathbf{c}_{skew}$ vectors.

The distributions of row and column energies using skewness measure are visualized in Fig. 5.2 using 2-dimensional embeddings generated using the t-SNE [290] algorithm. Fig. 5.2(a) shows t-SNE visualizations generated with the concatenated row and column skewness vectors computed from the MS. Similarly, Fig. 5.2(b) shows the visualization generated by concatenating the row and column skewness vectors computed from MHS and MPS, respectively. The representation for Mel-spectrogram has much overlap between the classes, even though it inherently contains both harmonic and percussive information. However, separately computing row and column skewness vectors from the MHS and MPS decompositions enhances the class separability, as can be observed in Fig. 5.2(b). The skewness values of HPSS based decomposition are observed to enhance the class separability of speech and music signals. Therefore, this decomposition might also be useful in detecting speech+music signals mixed at various SMR levels.

5.2.3 Multi-task learning framework

In a Single-Task Learning (STL) framework, separate models are trained for different problems. The STL models with sufficient parameters can approximate the underlying distribution very well. Moreover, these models perform reasonably well when learned from large enough datasets. For example, STL architectures proposed in [2, 212, 262, 263] have been successfully used for speech and music detection. However, in most practical cases, the performance of these models is constrained by the complexity of the underlying task and the generalizability to unseen data. On the other hand, an MTL framework attempts to overcome these problems by learning multiple closely related subproblems using a single model. Such a technique aids by learning the primary task through joint supervision of related auxiliary targets. This chapter explores the MTL framework for improving the detection performance of speech overlapped with music.

This chapter's main task is the 3-category classification of isolated speech, music, and overlapped speech+music. The traditional MTL framework and a cascaded-information variant of MTL are explored here. The proposed models designed in the traditional MTL framework involve simultaneous training of three auxiliary tasks (AT) that help learn the main task. First, a speech vs. non-speech classifier (AT_S) learns to differentiate between speech and non-speech. Music and speech+music are considered non-speech for AT_S. Second, the music vs. non-music classifier (AT_M) learns to discriminate between music and non-music. Here, speech and speech+music are considered non-music. The AT_S and AT_M are learned using a binary cross-entropy loss function. Third, a regression-based task (AT_R) tries to estimate the SMR of a given audio signal. The AT_R task is trained using a L_2 loss-based optimization scheme. The target output of the AT_R task, $\phi = [\phi_M, \phi_S]$, is a 2-dimensional vector that indicates the scaling factor of music (ϕ_M) with respect to speech (ϕ_S) in the input signal. Let the respective sets of music, speech, and speech+music signals are denoted by Γ_M, Γ_S, and Γ_{SM}. For a given input signal $x[n]$ and an SMR of v dB, the AT_R task target ϕ is computed using equation 5.7.

$$\phi = \begin{cases} [0,1]^T, & \text{if, } x[n] \in \Gamma_S \\ [1,0]^T, & \text{if, } x[n] \in \Gamma_M \\ [10^{-\frac{v}{10}}, 1]^T, & \text{if, } \{x[n] \in \Gamma_{SM}\} \wedge \{v \geq 0\text{dB}\} \\ [1, 10^{\frac{v}{10}}]^T, & \text{if, } \{x[n] \in \Gamma_{SM}\} \wedge \{v < 0\text{dB}\} \end{cases} \tag{5.7}$$

The proposed traditional MTL architecture is shown in Fig. 5.3(a). The hyper-parameters of each

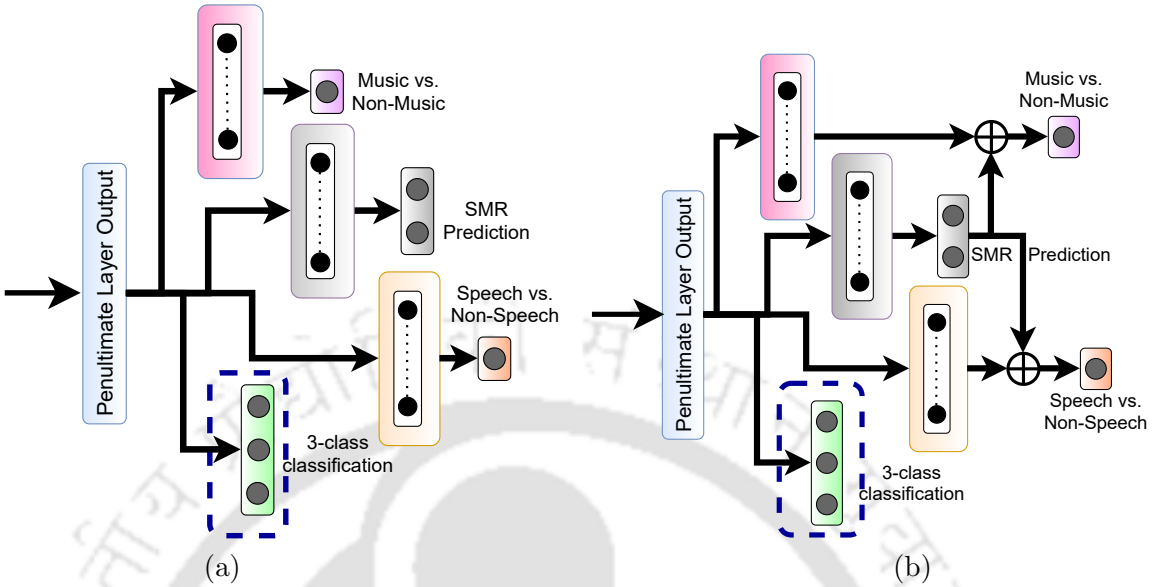


Figure 5.3: Illustrating the proposed design of (a) Traditional MTL and (b) Cascaded-information MTL architectures.

auxiliary sub-network have been tuned over a subset of the training data. The number of hidden layers are varied over $[1, 2, 3]$, while the number of hidden neurons are varied over $[16, 32, 64, 128]$. For the AT_S and AT_M tasks, both *Hinge* loss and *Binary-Crossentropy* loss were tested. The final tuned sub-networks of all the auxiliary tasks consist of a single fully connected hidden layer of 16 nodes with $ReLU$ activation. For regularization, the hidden layer is equipped with *Batch Normalization* and a *Dropout* fraction of 0.4. The output layer of the AT_S and AT_M tasks have a single neuron with *Sigmoid* activation function. The corresponding sub-networks are trained using *Binary-Crossentropy* loss function. The AT_R task has two nodes in its output layer with *Linear* activation and $l2$ loss function. The proposal for cascaded-information MTL variant is designed in a similar manner (shown in Fig. 5.3(b)). The 2-dimensional output from AT_R is concatenated with hidden layer outputs of the AT_S and AT_M tasks. The cascading of predicted SMR values is expected to aid the auxiliary tasks of speech vs. non-speech and music vs. non-music classification.

The proposed models use four separate loss functions. Let $\mathcal{L}_s$ be the loss function of AT_S, while y_s and $\hat{y}_s$ be its respective ground-truth and predicted outputs (equation 5.8). Let, $\mathcal{L}_m$ be the loss function for the AT_M task with respective true and predicted outputs as y_m and $\hat{y}_m$ (equation 5.9). The AT_R branch estimates the SMR proportion of speech and music in an input signal. The AT_R regression task is learned using a $l2$ loss $\mathcal{L}_{smr}$ (equation 5.10).

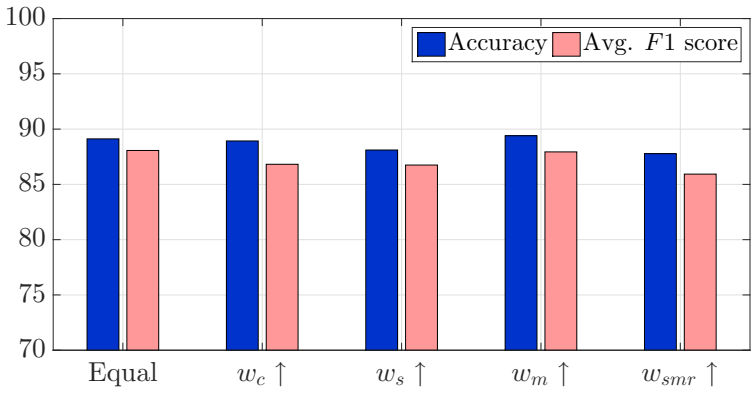


Figure 5.4: Illustrating the variations in classification performance obtained by using different loss weights for the main task and auxiliary tasks in the proposed MTL frameworks. Results are shown for the $B3$-MTL classifier with LMHPS-EF feature (see subsection 5.3.6).

$$\mathcal{L}_s = -\frac{1}{N_B} \sum_{i=0}^{(N_B-1)} (y_s[i] \log (\hat{y}_s[i]) + (1-y_s[i]) \log (1-\hat{y}_s[i])) \tag{5.8}$$

$$\mathcal{L}_m = -\frac{1}{N_B} \sum_{i=0}^{(N_B-1)} (y_m[i] \log (\hat{y}_m[i]) + (1-y_m[i]) \log (1-\hat{y}_m[i])) \tag{5.9}$$

$$\mathcal{L}_{smr} = \frac{1}{N_B} \sum_{i=0}^{(N_B-1)} ||y_{smr}[i]-\hat{y}_{smr}[i]||_2^2 \tag{5.10}$$

Here, $i = 0, \ldots (N_B - 1)$ are the samples in a training batch of size N_B. Also, y_{smr} and $\hat{y}_{smr}$ are the respective ground-truth and predicted values of the SMR proportion. The final 3-class classification loss function $\mathcal{L}_c$ for the main task is learned using a *Categorical-Crossentropy* loss function (equation 5.11).

$$\mathcal{L}_c = -\frac{1}{N_B} \sum_{i=0}^{(N_B-1)} \sum_{j=0}^{2} \left(y_c^{(j)}[i] log \left(\hat{y}_c^{(j)}[i] \right) \right) \tag{5.11}$$

Here, $y_c^{(j)}$ are the one-hot encoded ground truth and $\hat{y}_c^{(j)}$ are the predicted outputs of the j^{th} output neuron of the main task network. Here, $j = 0$ stands for music, $j = 1$ stands for speech, and $j = 2$ stands for speech+music. The total loss $\mathcal{L}_{Total}$ can be defined as the weighted sum of these four losses mentioned above. The MTL-based model is learned by minimizing $\mathcal{L}_{Total}$ (equation 5.12).

$$\mathcal{L}_{Total} = w_s.\mathcal{L}_s + w_m.\mathcal{L}_m + w_{smr}.\mathcal{L}_{smr} + w_c.\mathcal{L}_c \tag{5.12}$$

Table 5.1: Illustrating in detail the variations in classification performance obtained by using different loss weights for the main task and auxiliary tasks in the proposed MTL frameworks. Results are shown for the $B3$-MTL classifier with LMHPS-EF feature (see subsection 5.3.6).

			Music		Speech			Speech+Music			Avg.
Loss Weights	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
Equal	89.12 ±1.67	78.31 ±1.42	87.8 ±3.35	82.74 ±0.95	90.08 ±4.63	97.91 ±1.61	93.76 ±1.92	93.26 ±0.72	82.81 ±3.03	87.71 ±1.92	88.07 ±1.59
$w_c \uparrow$	88.93 ±0.81	71.59 ±2.53	90.77 ±0.71	80.03 ±1.7	92.56 ±1.66	96.63 ±1.42	94.54 ±0.58	93.45 ±0.29	79.46 ±2.33	85.88 ±1.34	86.82 ±0.78
$w_s \uparrow$	88.11 ±1.32	73.1 ±2.33	88.09 ±2.19	79.89 ±2	91.59 ±3.13	97.63 ±1.76	94.48 ±0.89	92.7 ±0.52	80.08 ±4.67	85.88 ±2.76	86.75 ±1.86
$w_m \uparrow$	89.40 ±2.46	75.99 ±5.12	90.29 ±1.64	82.48 ±3.56	91.6 ±2.48	96.79 ±0.59	94.1 ±1.11	93.44 ±1.15	81.91 ±5.99	87.24 ±3.89	87.94 ±2.83
$w_{smr} \uparrow$	87.78 ±0.42	73.37 ±1.42	87.34 ±0.93	79.75 ±1.13	88.56 ±1.64	98.25 ±0.6	93.15 ±0.82	92.83 ±0.72	78.21 ±1.34	84.89 ±0.69	85.93 ±0.19

The loss weights w_s, w_m, w_{smr} and w_c can be varied to obtain optimal loss minimization for a given task. The impact of each auxiliary task in the overall training of the classifier may also be understood by this process. Fig. 5.4 presents the performances obtained by assigning one task a higher weight while lower weights are assigned to the remaining tasks. In Fig. 5.4, "$w_c \uparrow$" indicates w_c=1, while the remaining weights are set to 0.5. Similar convention is followed for $w_s \uparrow$, $w_m \uparrow$, and $w_{smr} \uparrow$. The bars with x-label as "Equal" in Fig. 5.4 indicates that all loss weights are set to 1. It can be observed that the best performance is obtained when all losses are equally weighed. Even though the mean accuracy of $w_m \uparrow$ is slightly better than *Equal*, its standard deviation is more. Also, the average $F1$-score of *Equal* is better than $w_m \uparrow$. Therefore, all losses are equally weighed in this chapter. It can be concluded that every task in the proposed MTL framework is equally important and contributes toward better classifier training. Detailed performance measures are listed in Table 5.1. The results are shown for the best feature and classifier combination of LMHPS-EF and $B3$-MTL (see subsection 5.3.6).

5.3 Experiments and results

The proposed approach is validated using various experiments as described in this section. The music and speech data from the *MUSAN* - A Music, Speech, and Noise corpus [203] (≈ 102 hours) are used as experimental data. The *MUSAN* dataset is popularly used in a variety of speech and audio processing tasks, like speech music detection [267], general-purpose audio representation learning [291], music

relative loudness estimation [155], sound source separation works [292], speech enhancement [293] voice activity detection in the wild [294], and many others. For the initial experiments, data for the speech+music class is generated synthetically. However, in a later subsection (see subsection 5.3.10), results on real mixed speech and music signals from *DAFx-12* [206] and *Muspeak* [205] datasets are also reported to establish the efficacy of the proposed approach. This chapter uses three-fold cross-validation. Each experiment is run for three iterations, considering one of the folds as a test set and the remaining two folds as the training set. Results are reported using the mean and standard deviation computed over the three test runs.

The performance metrics used in this chapter are accuracy (Acc), precision (Prec), recall (Rec), and $F1$-score ($F1$). Let $CM_{3\times3}$ be the confusion matrix for a three-category classification evaluation, where rows indicate the true labels and columns indicate the predicted label. Accuracy measures the fraction of correct samples detected among all test data according to equation 5.13.

$$Acc = \frac{\sum_{a=1}^{3} CM[a][a]}{\sum_{a=1}^{3}\sum_{b=1}^{3} CM[a][b]} \tag{5.13}$$

Precision of a class determines how many predictions of that class truly belong to that class. The precision of the a^{th} class ($a = 0, 1, 2$) is computed using equation 5.14.

$$Prec[a] = \frac{CM[a][a]}{\sum_{b=1}^{3} CM[b][a]} \tag{5.14}$$

Recall of a class determines the fraction of true samples of that class correctly predicted. Recall of the a^{th} class ($a = 0, 1, 2$) is computed using equation 5.15.

$$Rec[a] = \frac{CM[a][a]}{\sum_{b=1}^{3} CM[a][b]} \tag{5.15}$$

$F1$-score for each class is computed as the harmonic mean of the precision and recall of that class. Thus, the $F1$-score of the a^{th} class ($a = 0, 1, 2$) is computed using equation 5.16.

$$F1[a] = \frac{2 \cdot Prec[a] \cdot Rec[a]}{Prec[a] + Rec[a]} \tag{5.16}$$

Finally, the average of class-wise $F1$-score (equation 5.17) is also reported.

$$\text{Avg. } F1 = \frac{1}{3}\sum_{a=1}^{3} F1[a] \tag{5.17}$$

All metrics are reported in percentage. The performance values are rounded to two decimal places. Hence, slight variations may be observed if the above equations are directly applied to the reported results. The process of generating the mixed signals is described next.

5.3.1 Synthetic speech+music signal generation

The MUSAN dataset contains ≈ 42 hours of music and ≈ 60 hours of speech. The available music and speech files are divided into three (almost equal) folds. Music files in the MUSAN dataset have genre annotations, while many speech files have gender information. Such available information was considered while grouping the files so that similar distribution of music and speech could be maintained across the folds. The speech+music data for each fold was created by mixing random pairs of music and speech files from the same fold. Files for mixing were chosen so that files from speech class (more in number) were sampled without replacement, while some files from the music class (less in number) were sampled at most twice. All integer SMR levels in the range −5dB to 20dB with a step of 1dB were considered for simulating real mixed signals. Here, SMR is defined by considering speech as the reference signal. It was ensured that for the speech+music data in each fold, an almost equal number of file pairs were mixed at each SMR level. The division of files from the *MUSAN* dataset into folds and speech+music file pairs with SMR annotations used in this chapter have been shared publicly (along with the codes[2]).

5.3.2 Baseline methods for comparison

The proposed approach is validated using four state-of-the-art speech music detection methods from the literature. First, the method proposed by Doukhan et al. [2] ($B1$) uses 21-Mel spectrogram input (MS) with a CNN to classify speech and music signals. The $B1$ classifier has four convolutional layers followed by four fully-connected layers (512 neurons each), leading to approximately 1.4 million parameters. Second, the proposal of Papakostas et al. [212] ($B2$) uses a CNN classifier with grayscale spectrogram input (S) to classify speech and music. The $B2$ classifier consists of three convolutional layers and two fully-connected layers (4096 neurons each), leading to approximately 44 million parameters. Third, Lemaire et al. [262] ($B3$) proposed a non-causal Temporal Convolution

[2] `https://github.com/mrinmoy-iitg/SM_HPSS_MTL`

Table 5.2: Illustrating the binary SMC performances of all the baseline systems $B1$, $B2$, $B3$, and $B4$ with their respective features.

				Music			Speech			Avg.
Feature	**Baseline**	**Context**	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
MS	$B1$	Original	94.86 ±2.04	94.41 ±3.81	94.08 ±1.09	94.22 ±2.09	95.90 ±0.87	95.95 ±3.01	95.92 ±1.77	95 ±1.73
S	$B2$	Original	93.99 ±1.25	93.19 ±2.47	92.25 ±0.42	92.71 ±1.43	94.69 ±0.59	95.31 ±1.89	95 ±1.21	93.67 ±1.53
LMS	$B3$	Original	*NOT COMPUTED*							
LS	$B4$	Original	92.28 ±5.91	89.05 ±9.61	92.86 ±4.54	90.84 ±6.95	94.75 ±3.64	91.8 ±7.74	93.21 ±5.61	92 ±6.08
MS	$B1$	695ms	94.86 ±2.04	94.41 ±3.81	94.08 ±1.09	94.22 ±2.09	95.90 ±0.87	95.95 ±3.01	95.92 ±1.77	95 ±1.73
S	$B2$	695ms	91.73 ±1.02	88.24 ±3.09	92.24 ±2.57	90.15 ±1.27	94.46 ±1.84	91.54 ±2.1	92.96 ±0.7	91.67 ±1.15
LMS	$B3$	695ms	96.5 ±0.42	96.77 ±0.92	95.38 ±1.1	96.06 ±0.47	96.87 ±0.5	97.78 ±0.71	97.32 ±0.21	96.69 ±0.32
LS	$B4$	695ms	90.12 ±7.88	83.57 ±14.43	94.86 ±1.27	88.45 ±9	95.95 ±1.36	86.12 ±13.57	90.44 ±8.41	89.33 ±8.96

Network (TCN) architecture with log-scaled 80-Mel spectrogram input (LMS) to detect speech and music in radio broadcasts. The $B3$ classifier consists of one TCN unit of three residual block stacks that add up to approximately 0.11 million parameters. The optimal $B3$ classifier was obtained by tuning the hyperparameters mentioned by the authors [262] on a subset of the training data used in this chapter (see subsection 5.3.1). Fourth, a CNN with 64 trainable Mel-scale convolutional filters with log-spectrogram input (LS) was proposed by Jang et al. [263] ($B4$) for music detection. With three convolution layers in addition to the Mel-scale one and two fully connected layers (2048 and 1024 neurons, respectively), the $B4$ classifier has approximately 21 million parameters to train. Among all the baselines considered, $B1$ uses the smallest and most challenging context window size of 695ms. Hence, all results reported in this work are computed for 695ms windows. However, the effect of varying context window size is also analyzed (see subsection 5.3.8). Binary SMC performance of the baseline methods is described next.

All the baselines were proposed as binary classification tasks. Table 5.2 presents the binary SMC performances of the baseline methods $B1$, $B2$, $B3$ and $B4$. The baseline results are computed with

Table 5.3: Optimization of the number of Mel filters (n_{mels}) for HPSS decomposition.

		Music			Speech			Speech+Music			Avg.
$\mathbf{n_{mels}}$	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
20	73.4 ±2.42	51.5 ±1.17	82.94 ±2.49	63.54 ±1.48	76.99 ±4.27	90.08 ±6.95	82.81 ±2.18	79.2 ±4.73	48.25 ±7.89	59.62 ±6.18	68.67 ±3.06
40	80.23 ±0.49	61.91 ±2.51	87.09 ±1.86	72.33 ±1.26	80.56 ±1	90.6 ±0.29	85.28 ±0.52	85.59 ±0.71	62.89 ±2.46	72.48 ±1.4	76.67 ±0.58
60	80.05 ±2.45	62.98 ±6.12	86.52 ±0.77	72.77 ±4.08	76.83 ±1.69	92.98 ±1.47	84.12 ±0.68	86.77 ±0.79	60.11 ±6.16	70.91 ±4.58	75.67 ±3.21
80	79.23 ±1.37	62.66 ±2.72	84.4 ±0.85	71.91 ±2	80.41 ±2.81	88.86 ±3.07	84.35 ±0.12	83.57 ±0.7	64.65 ±5.86	72.8 ±3.78	76.33 ±2.08
100	80.22 ±1.23	63.96 ±3.64	86.61 ±2.54	73.49 ±1.49	77.73 ±3.62	91.04 ±2.24	83.8 ±1.3	86.34 ±0.82	62.62 ±5.59	72.5 ±3.83	76.33 ±2.08
120	**81.75 ±1.01**	**63.92 ±3.26**	**87.86 ±0.54**	**73.97 ±2.14**	**80.91 ±4.65**	**90.95 ±1.84**	**85.54 ±1.87**	**86.88 ±0.57**	**64.83 ±8.06**	**74.04 ±5.3**	**77.67 ±3.21**

the respective features used in the original proposal. The $B1$ method used the MS feature with a context size of 695ms. The approach $B2$ used the S feature (see subsection 5.3.2) with a context size of 2400ms. The baseline $B3$ used the LMS feature with a context size of 90s. The method $B4$ used a 1010ms context of the LS feature. Apart from the original context sizes for different baselines, their performance at the smallest and most challenging 695ms context is also computed. Since the original context size of $B3$ is very long, its performance is only computed for 695ms. Short-term window size was fixed to 25ms, and frame size was kept at 10ms to maintain uniformity among all baselines. The speech and music data from the *MUSAN* dataset was used for this experiment. The mean and standard deviation of the performances of three-fold cross-validation are reported here. The previously described fold division is used here (see subsection 5.3.1). The training and test datasets used in the original work of the baselines and various experimental setups are different from this chapter. Nevertheless, comparable performances are obtained for all the baselines (Table 5.2). These baseline classifiers are extended to perform the three-category classification. This chapter aims to solve a three-category classification task. Here, the speech+music category has significant similarities with either the speech or music class based on the SMR. This increases the overall complexity of the classification task. Therefore, the three-category classification performance reported for the baselines might be lesser than their original binary classification performances.

Table 5.4: Optimization of the l_{harm} parameter for harmonic decomposition.

		Music			Speech			Speech+Music			Avg.
$\mathbf{l_{harm}}$	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
11	57.18 ±1.25	42.19 ±4.07	61.22 ±3.73	49.93 ±3.96	73.19 ±3.75	50.15 ±14.52	58.45 ±8.18	56.7 ±1.57	58.84 ±10.79	57.31 ±4.92	55.33 ±0.58
21	**57.99 ±0.98**	**44.68 ±3.67**	**60.19 ±5.83**	**51.28 ±4.53**	**71.05 ±3.02**	**55.53 ±7.66**	**62.09 ±4.87**	**58.3 ±2.7**	**59.47 ±5.36**	**58.82 ±3.49**	**57.4 ±2.02**
31	56.07 ±2.09	40.43 ±3.23	56.82 ±1.31	47.22 ±2.64	67.58 ±6.77	58.76 ±3.14	62.77 ±4.03	56.35 ±1.29	52.34 ±0.81	54.26 ±0.48	54.67 ±0.58
41	53.69 ±1.22	36.9 ±2.45	56.13 ±6	44.34 ±1.48	63.85 ±4.64	60.32 ±11.42	61.31 ±4.47	54.37 ±1.45	44.96 ±3.7	49.13 ±1.89	51.67 ±1.53
51	55.4 ±2.34	39.1 ±5.97	57.84 ±8.46	46.6 ±6.65	69.67 ±5.27	53.14 ±9.37	59.75 ±4.3	55.5 ±2.91	54.15 ±10.83	54.63 ±6.84	53.67 ±3.79

5.3.3 Experimental setup

The spectrograms in this chapter are computed using a short-term window size of 25ms and a frame size of 10ms. A heuristic-based energy threshold is used to remove silences in audio signals. The HPSS decomposition of spectrograms is performed using the *Librosa* [252] Python library. Classifiers are designed using the Keras [295] and Tensorflow [254] libraries. The following three hyperparameters are used in this chapter – the number of Mel filters (n_{mels}), the median filter size for harmonic decomposition (l_{harm}) and the median filter size for percussive decomposition (l_{perc}). The optimal values of hyperparameters were obtained experimentally. The experiments were performed over a subset of the training set. The $B3$ classifier was used for this experiment. The values of n_{mels}, l_{harm} and l_{perc} were respectively varied over $[20, 40, 60, 80, 100, 120]$, $[11, 21, 31, 41, 51]$ and $[11, 21, 31, 41, 51]$. The optimal values obtained after the tuning experiments are $n_{\text{mels}} = 120$, $l_{\text{harm}} = 21$, and $l_{\text{perc}} = 11$. The detailed results obtained for these experiments are listed in Tables 5.3, 5.4, and 5.5. The classifiers are trained using feature patches with $n_t = 68$ frames (695ms) as the temporal context. Patches are extracted with a shift of 68 frames. The models are trained with a minibatch size of 48 and a maximum of 50 epochs. An early-stopping criterion has been used while training to avoid overfitting. Early-stopping is a regularization approach that monitors the validation loss and terminates the model training if there is no improvement for consecutive 5 epochs. The best model with the lowest validation loss obtained in the process is retained. All codes used for performing experiments in this chapter have

Table 5.5: Optimization of the l_{perc} parameter for percussive decomposition.

		Music			Speech			Speech+Music			Avg.
$\mathbf{l_{perc}}$	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
11	**58.75 ±2.72**	**43.44 ±4.05**	**62.02 ±2.19**	**51.01 ±2.75**	**69.79 ±6.03**	**58.2 ±10.1**	**62.92 ±4.69**	**58.63 ±2.35**	**55.56 ±12.7**	**56.71 ±7.58**	**57 ±2.65**
21	58.25 ±3.21	42.97 ±5.97	60.92 ±3.6	50.08 ±2.62	69.35 ±2.46	55.14 ±8.23	61.29 ±6.17	58.28 ±2.43	56.74 ±7.26	57.38 ±4.53	56.33 ±3.51
31	56.47 ±1.47	42.09 ±4.59	54.83 ±6.66	47.61 ±5.36	71.53 ±1.41	56 ±5.65	62.7 ±3.32	57.12 ±1	59.11 ±2.36	58.09 ±1.48	56 ±1.73
41	55.91 ±2.1	43.4 ±3.38	54.51 ±2.29	48.29 ±2.59	75.81 ±5.29	50.7 ±2.93	60.66 ±2.25	56.14 ±1.61	63.44 ±4.58	59.54 ±2.82	56.33 ±2.52
51	57.23 ±3.19	42.56 ±8.28	60.8 ±3.59	49.69 ±5.86	74.27 ±4.59	49.88 ±8.31	59.4 ±6.6	56.6 ±4.86	59.75 ±12.73	57.84 ±8.26	55.67 ±5.69

been shared publicly (see subsection 5.3.1). The results are discussed in the following subsections.

5.3.4 Performance of Harmonic-Percussive features

The 3-class classification performance of $B1$, $B2$, $B3$ and $B4$ are tabulated in Table 5.6. The best average $F1$-score of 76.33 ±2.08 is obtained for the baseline $B3$ with the LMS feature. The $B4$ baseline with LS input seems to perform the poorest. However, it was observed that the $B4$ model overfits the training data. This might be attributed to the large model size of $B4$. Reducing the number of parameters by removing the fully-connected (FC) layers in the $B4$ architecture greatly improved its performance (see $B4$ (NoFC) in Table 5.6). The best performing $B3$ model is used in further experiments for developing the best feature and classifier combination. Later, performance improvements obtained with all baselines using the proposed methods are described in subsection 5.3.6.

Performance of the proposed HPSS decomposed features in the current 3-class classification task with the best baseline $B3$ is tabulated in Table 5.7. The performance of $B3$ was further improved by setting n_{mels}=120 (first row in Table 5.7). The performance of $B3$ with log-scaled 120-Mel harmonic spectrogram input (LMHS) computed with tuned parameter l_{harm}=21 is listed in the second row. In the third row, the performance of $B3$ with log-scaled 120-Mel percussive spectrogram input (LMPS) computed with tuned l_{perc}=11 is provided. In the last row, the performance of $B3$ with the early-fusion (EF) of the LMHS and LMPS features concatenated along the feature dimension (LMHPS-EF) is listed. The EF might be the best among all fusion strategies explored in this chapter (see subsec-

Table 5.6: Illustrating the performances of baseline methods used for comparisons in this chapter.

Feature	Classifier	Music				Speech			Speech+Music			Avg.
		Acc	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
MS	$B1$	71.68 ±2.28	49.5 ±3.37	77.64 ±11.28	60.19 ±4.29	81.03 ±1.35	87.83 ±6.03	84.19 ±2.21	75.18 ±4.66	51.86 ±4.18	61.32 ±3.9	68.67 ±3.06
S	$B2$	67.39 ±2.02	40.26 ±3.08	61.4 ±8.82	48.59 ±5.04	78.31 ±4.4	93.99 ±4.3	85.38 ±3.52	74.85 ±1.18	47.36 ±5.25	57.88 ±3.86	64 ±2.65
LMS	$B3$	79.23 ±1.37	62.66 ±2.72	84.4 ±0.85	71.91 ±2	80.41 ±2.81	88.86 ±3.07	84.35 ±0.12	83.57 ±0.7	64.65 ±5.86	72.8 ±3.78	76.33 ±2.08
LS	$B4$	56.86 ±10.41	67.45 ±12.02	46.22 ±29.72	49.16 ±18.32	90.92 ±11.82	40.98 ±33.76	48.58 ±36.4	59.19 ±7.47	84.29 ±9.61	69.04 ±4.81	55.67 ±13.32
LS	$B4$ (NoFC)	69.95 ±16.97	48.59 ±13.58	88.06 ±11	61.09 ±9.6	89.02 ±8.22	63.05 ±46.68	63.93 ±39.83	81.08 ±18.57	59.67 ±15.59	68.01 ±14.05	64.35 ±20.52

tion 5.3.7). It can be observed that the LMHS feature does not perform better than LMS. A possible reason might be that Mel-scaling reduces the resolution of high-frequency harmonics that hampers discrimination. The LMPS provides almost similar results to LMS, although with a lower standard deviation. However, the LMHPS-EF significantly improves the baseline LMS performance (average $F1$-score) by around 7%. The LMHPS-EF feature also performs better for each of the individual classes. Such performances support the proposal of this chapter that the features generated from the HPSS decomposition of the spectrogram are efficient in detecting speech, music, and speech+music signals.

5.3.5 Performance of MTL framework

The second contribution of this chapter is an exploration of the popular MTL framework in the current task. This chapter explores the traditional MTL architecture [277] and a cascaded-information MTL variant [287] (see subsection 5.2.3). Table 5.8 lists the performances of the best baseline classifier $B3$ whose architecture is modified according to traditional MTL framework ($B3$-MTL) and the cascaded-information MTL framework ($B3$-C-MTL), as shown in Fig. 5.3. With the $B3$-MTL architecture, the performance of the baseline LMS feature improves by around 2%. Similarly, the performances of proposed features LMHS, LMPS, and LMHPS-EF improve by around 6%, 5%, and 3%, respectively. The $B3$-C-MTL architecture also improves the performances of LMS, LMHS, LMPS, and LMHPS-EF

Table 5.7: Illustrating the effect of using the optimized number of Mel-filters along with Harmonic and Percussive features with the best performing baseline ($B3$). Here, n_{mels}=120, l_{harm}=21, and l_{perc}=11 are used as the tuned parameters.

Feature	Classifier	Music				Speech			Speech+Music			Avg.
		Acc	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
LMS	$B3$	81.75 ±1.01	63.92 ±3.26	**87.86 ±0.54**	73.97 ±2.14	80.91 ±4.65	90.95 ±1.84	85.54 ±1.87	86.88 ±0.57	64.83 ±8.06	74.04 ±5.30	77.67 ±3.21
LMHS	$B3$	79.02 ±5.02	52.61 ±8.02	87.14 ±1.77	65.43 ±6.63	83.59 ±4.59	**97.11 ±1.25**	89.78 ±2.10	88.91 ±2.74	52.24 ±15.48	65.12 ±12.55	73.33 ±6.81
LMPS	$B3$	81.77 ±1.31	56.48 ±3.49	86.29 ±3.20	68.17 ±1.66	89.95 ±5.88	95.26 ±4.03	92.36 ±1.87	89.06 ±1.26	64.16 ±4.11	74.52 ±2.67	78.33 ±1.53
LMHPS-EF	$B3$	**86.87 ±1.38**	**70.67 ±5.06**	86.85 ±4.38	**77.74 ±1.18**	**90.47 ±3.10**	97.10 ±1.13	**93.64 ±1.41**	**92.06 ±2.13**	**77.82 ±6.06**	**84.20 ±2.67**	**85.19 ±1.75**

features by around 2%, 9%, 4%, and 2%, respectively. However, the overall best average $F1$-score of 88.07 ±1.59 for the 3-class classification task is obtained with the $B3$-MTL architecture with the LMHPS-EF feature. Performances of all three classes also improve significantly with the LMHPS-EF feature and $B3$-MTL classifier. Hence, it can be inferred that the use of the MTL framework in the current task helps in learning more generalizable representations for distinguishing the different audio classes, thereby significantly improving the overall performance.

5.3.6 HPSS features and MTL framework with baselines

The improvements obtained for $B3$ with the usage of harmonic-percussive features and MTL-based classifier modification motivate the application of these changes to other baselines. Each baseline is fed with the EF of harmonic and percussive features with the respective preprocessing of each baseline. Moreover, all the baselines are equipped with the best-performing MTL modification. Thus, $B1$ is modified to $B1$-MTL and provided the EF of 120-Mel harmonic and percussive spectrograms (MHPS-EF) as input. The $B2$ is modified to $B2$-MTL and given the EF of harmonic and percussive spectrograms (HPS-EF) as input. Finally, $B4$ is converted to $B4$-MTL and trained with EF of log-scaled harmonic and percussive spectrograms (LHPS-EF) as input. Table 5.9 shows the improved performances of all the baselines used in this chapter. It can be observed that there are significant improvements to the performances of all the baselines. The $B3$-MTL classifier with the LMHPS-

Table 5.8: Illustrating the effect of modifying the best performing baseline ($B3$) with traditional MTL-based and cascaded-information MTL-based (C-MTL) frameworks, as shown in Fig. 5.3. The C-MTL modification helped improve the performance of LMS and LMHS features. However, the best result with the LMHPS-EF feature was obtained with the traditional MTL modification.

Feature	Classifier		Music			Speech			Speech+Music			Avg.
		Acc	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
LMS	$B3$-MTL	81.79 ±2.59	68.05 ±4.06	85.18 ±2.72	75.63 ±3.28	82.33 ±3.52	89.99 ±2.23	85.98 ±2.82	85.43 ±2.59	70.48 ±3.15	77.22 ±2.74	79.33 ±2.52
LMHS	$B3$-MTL	83.27 ±1.08	62.52 ±7.52	83.51 ±1.9	71.28 ±4.39	85.86 ±1.16	96.65 ±1.02	90.93 ±0.77	88.87 ±0.76	67.77 ±5.78	76.82 ±3.86	79.67 ±2.52
LMPS	$B3$-MTL	85.44 ±0.08	64.76 ±2.78	84.87 ±1.43	73.42 ±1.29	91.65 ±2.95	97.35 ±0.6	94.39 ±1.38	90.8 ±0.62	74.41 ±0.33	81.79 ±0.37	83.2 ±0.24
LMHPS-EF	$B3$-MTL	89.12 ±1.67	78.31 ±1.42	87.8 ±3.35	82.74 ±0.95	90.08 ±4.63	97.91 ±1.61	93.76 ±1.92	93.26 ±0.72	82.81 ±3.03	87.71 ±1.92	88.07 ±1.59
LMS	$B3$-C-MTL	82.72 ±1.72	67.32 ±7.23	87.23 ±1.6	75.8 ±4.01	84.1 ±0.99	88.57 ±2.79	86.25 ±0.94	85.45 ±1.82	70.87 ±4.63	77.46 ±3.48	79.67 ±2.52
LMHS	$B3$-C-MTL	84.62 ±1.97	66.66 ±5.64	84.56 ±2.27	74.45 ±3.68	87.65 ±3.98	96.08 ±1.54	91.61 ±1.55	89.92 ±0.35	73.16 ±6.15	80.59 ±3.94	82.33 ±3.06
LMPS	$B3$-C-MTL	85.11 ±1.99	64.84 ±5.07	83.12 ±1.46	72.74 ±2.64	90.51 ±3.31	97.62 ±0.8	93.91 ±1.84	90.06 ±0.4	73.83 ±5.94	81.06 ±3.65	82.67 ±2.52
LMHPS-EF	$B3$-C-MTL	90.09 ±0.66	74.05 ±2.49	90.8 ±1.99	81.54 ±0.75	91.66 ±2.11	97.54 ±0.74	94.49 ±0.79	94.28 ±1.28	80.71 ±1.64	86.95 ±0.51	87.33 ±0.58

EF feature performs the best, while the $B2$-MTL provides a minor improvement. Nevertheless, the observed results validate the current proposal that HPSS based features and MTL-framework can be an efficient combination for detecting music, speech, and speech+music.

Table 5.10 illustrates the performance of all the four baseline methods on the speech, music and speech+music classes in the *Movie-MUSNOMIX* dataset (see section 3.2). It may be observed that the performance of all the baseline methods improves upon adding the harmonic and percussive features and the MTL-based modifications to their classifiers. The $B1$-MTL and $B3$-MTL systems provide significant performance gains. Such results also validate the proposal of this chapter on challenging movie audio signals as well.

Table 5.9: Illustrating the improvement in performances of all four baselines with the Harmonic-Percussive feature and MTL modification of their respective classifier architectures.

Feature	Classifier	Music				Speech			Speech+Music			Avg.
		Acc	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
MHPS-EF	*B*1-MTL	**89.67** ±**2.92**	67.03 ±6.92	**96.27** ±**2.43**	78.84 ±4.5	**94.36** ±**2.76**	**99.25** ±**0.98**	**96.72** ±**1.28**	**97.4** ±**2.06**	74.54 ±8.05	84.23 ±4.81	86.6 ±3.51
HPS-EF	*B*2-MTL	69.60 ±1.89	45.79 ±0.39	63.77 ±9.23	53.14 ±3.24	79.47 ±5.7	90.27 ±3.98	84.35 ±1.65	74.71 ±2.05	54.87 ±0.48	63.27 ±1.04	66.92 ±1.71
LMHPS-EF	*B*3-MTL	89.12 ±1.67	**78.31** ±**1.42**	87.8 ±3.35	**82.74** ±**0.95**	90.08 ±4.63	97.91 ±1.61	93.76 ±1.92	93.26 ±0.72	**82.81** ±**3.03**	**87.71** ±**1.92**	**88.07** ±**1.59**
LHPS-EF	*B*4-MTL (NoFC)	73.8 ±6.93	59.03 ±7.29	63.3 ±22.54	58.89 ±6.18	83.88 ±4.45	87.57 ±12.61	85.31 ±6.24	77.27 ±10.43	69.68 ±10.35	72.3 ±2.18	72.17 ±4.04

5.3.7 Feature fusion strategies

Three feature fusion strategies for combining the harmonic and percussive features are compared in this chapter. The *B*3-MTL classifier is used in this experiment as it has performed the best so far. First, the Early-Fusion (EF) strategy involves concatenating LMHS and LMPS features along the feature dimension. With this strategy, the number of trainable parameters for the *B*3-MTL classifier increases by approximately 3K. Second is the Intermediate-Fusion (IF) strategy, where the initial convolution layers of *B*3-MTL are kept separate for both the features. Flattened embeddings obtained from each convolutional sub-networks are concatenated and passed on for classification. The intermediate fusion (IF) of LMHS and LMPS features requires training of about additional 0.22 million parameters. The third is the Late-Fusion (LF) strategy, where two separate models are trained for LMHS and LMPS. Predictions for the same test signal from both models are combined linearly for the final decision during testing. Let, p_{harm} and p_{perc} be the predictions from both the models. The LF decision score is obtained using Eqn 5.18.

$$\alpha_{LF} \cdot p_{\text{harm}} + (1 - \alpha_{LF}) \cdot p_{\text{perc}} \tag{5.18}$$

The trainable parameters become twice that of a single *B*3-MTL classifier with the LF strategy. Table 5.11 lists the performances obtained for each of the fusion strategies discussed above. The best

Table 5.10: Illustrating the performance of the HPSS based features and MTL-based classifiers on the speech, music and speech+music classes of the *Movie-MUSNOMIX* dataset.

Feature	Classifier	Music				Speech			Speech+Music			Avg.
		Acc	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
MS	$B1$	84.21 ±0.48	93.99 ±0.54	91.76 ±1.62	92.86 ±0.94	71.61 ±4.12	86.31 ±2.06	78.16 ±1.8	85.46 ±3.39	75.37 ±2.94	80.0 ±1.28	83.67 ±1.34
S	$B2$	71.93 ±1.55	73.31 ±3.6	90.79 ±3.1	81.09 ±3.21	63.74 ±4.15	73.65 ±1.66	68.2 ±1.76	72.06 ±4.89	53.39 ±3.71	61.22 ±3.14	70.17 ±2.70
LMS	$B3$	77.75 ±3.7	81.88 ±0.9	87.46 ±3.07	84.56 ±1.64	68.41 ±6.45	80.86 ±3.12	73.99 ±4.44	76.29 ±6.16	63.74 ±7.38	69.42 ±6.84	75.99 ±4.31
LS	$B4$	80.69 ±1.22	75.36 ±5.23	86.63 ±5.68	80.23 ±0.77	76.87 ±8.51	86.65 ±7.21	80.75 ±3.04	82.53 ±5.78	65.61 ±8.11	72.6 ±4.38	77.86 ±2.73
MHPS-EF	$B1$-MTL	90.87 ±0.31	90.9 ±3.61	97.7 ±0.77	94.15 ±2.14	82.18 ±5.58	94.17 ±3.04	87.59 ±2.69	94.78 ±1.9	80.87 ±1.52	87.25 ±0.51	89.66 ±1.78
HPS-EF	$B2$-MTL	74.64 ±1.06	77.49 ±4.1	89.99 ±2.64	83.23 ±3.15	60.99 ±3.89	78.78 ±2.98	68.58 ±1.43	76.47 ±4.59	53.62 ±3.94	62.85 ±2.4	71.55 ±2.33
LMHPS-EF	$B3$-MTL	81.93 ±3.01	82.53 ±2.66	88.4 ±1.38	85.36 ±2.05	73.93 ±3.43	89.6 ±0.4	80.98 ±2.12	83.33 ±2.85	67.84 ±5.99	74.74 ±4.78	80.36 ±2.98
LHPS-EF	$B4$-MTL (NoFC)	78.54 ±0.07	82.22 ±7.01	79.25 ±11.63	79.64 ±2.86	78.24 ±5.17	83.36 ±7.0	80.27 ±0.95	76.61 ±2.42	72.92 ±7.06	74.41 ±2.37	78.11 ±2.06

performance for the LF strategy is obtained with $\alpha_{LF} = 0.5$. The LF strategy does improve upon the individual performances of LMHS and LMPS with the $B3$-MTL classifier (listed in Table 5.8), indicating that the two features carry some complementary information. However, the best average $F1$-score is obtained with the EF strategy. A possible reason for the poor performance observed with the IF strategy might be the confusion created between music and speech+music signals. With the IF strategy, the precision of the music class and the recall of the speech+music class are significantly reduced (see Table 5.11), indicating that speech+music is frequently detected as music.

5.3.8 Effect of context window size

The size of the context window is supposed to affect the performance of classifiers. Hence, the best feature and classifier combination of LMHPS-EF with $B3$-MTL is trained with different context window sizes, 250ms, 500ms, 750ms, 1000ms and 2500ms. Fig. 5.5 shows these results. The perfor-

Table 5.11: Illustrating the results of employing various feature fusion strategies, viz., Early-Fusion (EF), Intermediate-Fusion (IF) and Late-Fusion (LF) with the best classifier $B3$-MTL. The best LF performance was obtained for $\alpha_{LF} = 0.5$.

		Music			Speech			Speech+Music			Avg.
Feature	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**F1**
LMHPS-EF	89.12 ±1.67	78.31 ±1.42	87.8 ±3.35	82.74 ±0.95	90.08 ±4.63	97.91 ±1.61	93.76 ±1.92	93.26 ±0.72	82.81 ±3.03	87.71 ±1.92	88.07 ±1.59
LMHPS-IF	90.86 ±2.27	65.23 ±7.1	90.02 ±2	75.52 ±4.99	89.97 ±6.85	94.78 ±2.92	92.13 ±2.37	91.47 ±0.7	71.91 ±10.67	80.19 ±6.54	82.62 ±4.51
LMHPS-LF	88.23 ±1.37	70.21 ±6.05	89.14 ±0.76	78.44 ±3.55	91.36 ±2.6	97.94 ±0.8	94.52 ±1.3	93.53 ±0.32	77.61 ±2.54	84.82 ±1.63	85.92 ±1.41

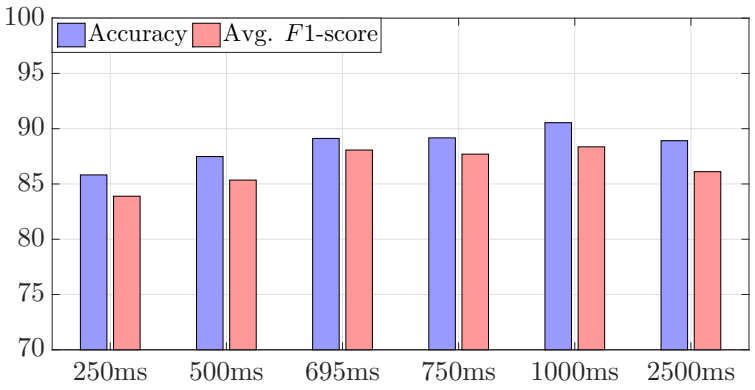


Figure 5.5: Figure depicting the effect of different context window sizes used for making the classification decision with the LMHPS-EF feature and $B3$-MTL classifier. It can be observed that performance improves with a larger context but tends to stabilize with sizes greater than 695ms.

mance improves with increasing the context window size to 1000ms. The improvement in $F1$-score is significant from 250ms to 500ms and 750ms. However, the $F1$-score does not improve significantly from 750ms to 1000ms and drops from 1000ms to 2500ms. Performance for the 695ms context window (used for experiments in the manuscript) is slightly better than that at 750ms. In general, the performance improves with increasing context window sizes. However, beyond 695ms the performances tend to oscillate around 89.54 accuracy (or 87.39 Avg. $F1$-score). Thus, it may be inferred that increasing the context window size beyond a limit may not add any extra benefit to the underlying task. Detailed performance measures are presented in Table 5.12

Table 5.12: Illustrating the effect of different context window sizes used for making the classification decision with the LMHPS-EF feature and $B3$-MTL classifier. It can be observed that the performance improves with increasing context window till 695ms but tends to oscillate around 89.54 accuracy (or 87.39 Avg. $F1$-score) beyond 695ms.

Context		Music			Speech			Speech+Music			Avg.
(in ms)	Acc	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	F1
250	85.82 ±0.99	67.42 ±2.87	85.41 ±1.93	75.31 ±1.04	90.7 ±1.36	97.05 ±0.32	93.76 ±0.6	91.03 ±0.37	75.71 ±4.68	82.61 ±2.67	83.89 ±1.41
500	87.48 ±0.42	71.42 ±2.05	86.18 ±1.59	78.08 ±0.61	90.12 ±3.21	97.38 ±1.62	93.57 ±1.48	91.64 ±0.35	78.24 ±3.44	84.38 ±1.88	85.35 ±1.28
695	89.12 ±1.67	78.31 ±1.42	87.8 ±3.35	82.74 ±0.95	90.08 ±4.63	97.91 ±1.61	93.76 ±1.92	93.26 ±0.72	82.81 ±3.03	87.71 ±1.92	88.07 ±1.59
750	89.17 ±0.81	74.95 ±2.83	88.5 ±1.16	81.13 ±1.19	92.19 ±1.3	97.47 ±1.17	94.75 ±0.59	93.14 ±0.58	82.05 ±1.84	87.23 ±0.82	87.7 ±0.44
1000	90.54 ±0.34	76.47 ±0.75	90.66 ±0.57	82.96 ±0.23	91.39 ±3.04	97.73 ±1.74	94.41 ±0.97	94.25 ±0.5	82.06 ±2.43	87.72 ±1.17	88.36 ±0.61
2500	88.91 ±2.05	69.72 ±2.61	90.68 ±2.66	78.83 ±2.66	92.26 ±1.47	97.22 ±1.37	94.67 ±1.32	93.49 ±2.42	77.61 ±2.14	84.81 ±2.22	86.11 ±1.99

5.3.9 Performance at challenging SMR levels

An important goal of this chapter is the detection of speech+music signals in challenging SMR scenarios. All performances reported till now are computed for speech+music signals mixed at different SMR levels in the aforementioned range $[-5, \ldots 20]$ dB. It has been encouraging to observe that the current proposal performs quite well in the presence of mixed signals at various SMR levels. However, it is also important to assess the capability of the proposed approach in detecting speech+music signals at specific challenging SMR levels. In this context, results are computed at -5dB, 0dB, 5dB, 10dB, 15dB, and 20dB. Here, -5dB and 20dB are the most challenging cases since the music component is 5dB louder than the speech component in the former, while speech is 20dB louder in the latter. In such cases, the speech+music signal is expected to be confused with the louder component. The performance in such cases will highlight the robustness of the proposed approach. Just for this experiment, the SMR annotations that were fixed for every speech+music signals (see subsection 5.3.1) were substituted with the particular SMR level being tested (among $[-5, 0, 5, 10, 15, 20]$ dB). The mean recall of over 3-folds for detecting speech+music signals is reported.

Fig. 5.6 compares the performances of all the baseline classifiers with their proposed modifications.

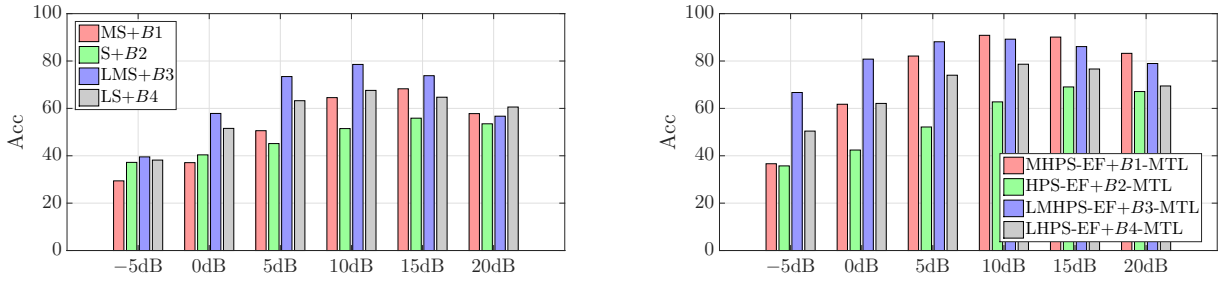


Figure 5.6: This figure illustrates the performance of (a) baseline models and (b) their modified versions at varying SMR levels. Accuracy values are reported in percentage.

The obtained results indicate that the performances of all baselines improve with the proposed modifications at all six chosen SMR levels. The most significant improvements for the $B1$ and $B3$ baselines can be observed. The performances of all systems peak around 10dB and expectedly drop towards the challenging SMR cases. The $B3$ model with the proposed modifications performs much better than the others at challenging SMRs. Also, the performance with louder music (at −5dB) is poorer than with louder speech (at 20dB) for all baseline systems. Such a result indicates that the trained models are confused more in the presence of loud music than speech. This observation might be attributed to the fact that speech is a relatively low-frequency signal [296] when compared to music. Thus, only a limited range of frequencies might be dominated by loud speech in a speech+music signal, enabling better detection. In comparison, loud music might be dominating a more extensive spectral range, thereby creating more confusion. Nonetheless, the overall performances obtained at challenging SMR levels establish that the current proposal is effective.

5.3.10 Performance with real mixed signals

The performances reported so far are computed over synthetically generated speech+music signals. However, the efficacy of the proposed approach can be ascertained when tested with real speech and music signals present as isolated and overlapping mixtures. Schlüter et al. [206] created a dataset (*DAFx-12* dataset) of recorded Swiss and Austrian radio broadcasts. The authors manually annotated the recordings into speech/non-speech and music/non-music segments. The *DAFx-12* dataset consists of around 28 hours of music, 8 hours of speech, and 5 hours of speech+music segments. The dataset is divided into a training set of around 15 hours, a validation set of 6 hours, two test sets of 9 hours (Swiss recordings), and 12 hours (Austrian recordings). The reader is encouraged to refer [206] for

Table 5.13: Performance of the proposed approach on real signals from the *DAFx-12* dataset [206] are tabulated here. Baseline results are quoted directly from the reference. Here, *Mu/Non-Mu* indicates Music vs. Non-Music detection performance, *Sp/Non-Sp* indicates Speech vs. Non-Speech detection performance, and *SpMu/Non-SpMu* indicates Speech+Music vs. Non-Speech+Music detection performance.

Method	Test set	Mu/Non-Mu				Sp/Non-Sp				SpMu/Non-SpMu			
		Acc	**Prec**	**Rec**	**F1**	**Acc**	**Prec**	**Rec**	**F1**	**Acc**	**Prec**	**Rec**	**F1**
Schlüter et al. [206]	Swiss	97.30	98.80	98.00	98.40	98.40	96.40	96.50	96.40	–	–	–	–
	Austrian	95.60	95.30	97.40	97.30	97.00	95.90	95.10	95.50	–	–	–	–
Proposed	Swiss	96.37	97.53	**98.20**	97.86	97.58	95.08	94.37	94.72	95.04	71.39	64.86	67.97
	Austrian	93.68	94.96	**97.48**	96.21	95.91	94.55	93.36	93.95	91.19	74.12	73.03	73.57

Table 5.14: Event-level performance on Muspeak dataset [205]. The onset-offset $F1$-score at different tolerance durations is reported.

		Dataset	**F1-score** (500ms)	**F1-score** (1000ms)
Music	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.0930	0.1142
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.2235	0.2480
	LMHPS-EF + *B*3-MTL	Muspeak [205]	0.2870 ±0.0210	0.3069 ±0.0224
Speech	Doukhan et al. [2]	MIREX Evaluation Dataset 1	0.1603	0.2122
	Doukhan et al. [2]	MIREX Evaluation Dataset 2	0.4139	0.4350
	LMHPS-EF + *B*3-MTL	Muspeak [205]	0.4017 ±0.1626	0.4604 ±0.1908

more details about the *DAFx-12* dataset.

Schlüter et al. [206] trained two separate classifiers to detect speech and music separately. Following their approach for a fair comparison, two separate classifiers are trained in this chapter to evaluate the proposed approach on the *DAFx-12* dataset. For generalization purposes, silence removal is

not performed for the *DAFx-12* dataset. The $B3$-MTL model trained on the LMHPS-EF feature over 695ms context is used in this experiment in a transfer learning mode. For the music detection classifier ($B3$-MTL-Mu), except for the music/non-music output, others are stripped off from the $B3$-MTL model. The remaining weights in the $B3$-MTL-Mu model are initialized with those from the trained $B3$-MTL model. The weights are subsequently tuned over the training set of the *DAFx-12* dataset. Similarly, all but the speech/non-speech output are stripped off from the $B3$-MTL model to create the speech detection classifier ($B3$-MTL-Sp). The weights of $B3$-MTL-Sp are initialized and tuned in a similar manner as $B3$-MTL-Mu. Both the models are tuned with the Nadam optimizer [297] with an initial learning rate of 2×10^{-3}. The previously mentioned early-stopping criterion is also used here (see subsection 5.3.3).

Table 5.13 lists the results of the proposed method on the *DAFx-12* dataset. The baseline results [206] are directly quoted from the paper. It can be observed that the proposed approach provides performances comparable with the baseline for both the test sets. For the music/non-music detection, the proposed method provides a slightly better recall as well. The results for speech+music detection in Table 5.13 are generated by combining the predictions from both the $B3$-MTL-Mu and $B3$-MTL-Sp classifiers. The proposed approach provides a reasonable $F1$-score for detecting speech+music as well. However, small differences in the music and speech detection performances are observed that can be reasoned as follows.

First, the baseline result was computed using approximately 46ms frames with a shift of around 23ms and a context window of ≈923ms. The proposed approach uses a frame-size, frame-shift, and context window of 25ms, 10ms, and 695ms, respectively. Second, the model trained on the *MUSAN* dataset is transfer-learned on the *DAFx-12* dataset in this work. Whereas, the model of Schlüter et al. [206] is trained from scratch on the *DAFx-12* dataset. Despite the slight differences, the results obtained are encouraging. It can be concluded that the proposed method of using harmonic-percussive spectrogram decomposition with the MTL framework can be an effective method of detecting not only isolated speech and music signals but also their mixtures.

Another dataset consisting of continuous sequences of speech and music signals is the *Muspeak* dataset. This dataset was used as a training dataset in the MIREX challenges. The performance of the proposed approach is also validated on the *Muspeak* dataset. The $B3$-MTL-Mu and $B3$-MTL-Sp models tuned on the *DAFx-12* dataset are used in this experiment. The trained classifiers are

used to predict the speech and music sequences for the audio files in the *Muspeak* dataset without any additional training. The obtained results are listed in Table 5.14. The baseline performances of the best performing system of Doukhan et al. [2] in Table 5.14 are provided just as a reference. The baseline results are computed on unreleased evaluation data of the MIREX challenge and can only indicate the scale of state-of-the-art performance. In this context, the obtained performances for music and speech detection are promising. The proposed system performs better in detecting speech than music. Such a performance also endorses the generalizability of classifiers trained in the MTL framework. This validates the current proposal of using HPSS features and MTL framework-based classifiers on real-world speech and music signals as well.

5.4 Summary

This chapter proposes the use of harmonic-percussive source separation (HPSS) to generate features that are found to be better suited for detecting speech+music signals mixed at varying SMR levels. The baseline classifiers were modified in the traditional and cascaded-information multi-task learning (MTL) framework to improve classification performance. The HPSS features are found to perform better than baseline features. Results reported for different feature fusion strategies indicate that early fusion performs best in the current scenario. The use of the MTL framework also aids in further improvement of the performances. Results are reported over both synthetic speech+music data generated using the *MUSAN* dataset and real mixed data from the *DAFx-12* [206] and *Muspeak* [205] datasets.

Chapters 3, 4 and 5 have focused on detecting speech and music signals either in isolated or in overlapped conditions. Speech and music signals constitute a significant portion of the audio of movies. Automatic genre classification of movies is an important task from the content analysis aspect. However, processing whole movie audio might not be computationally feasible. Movie creators provide trailer videos that are a concise compilation of the important events in a movie [36]. Trailers help the audiences to build an idea about what can be expected from the movie. It is observed that the amount and distribution of speech and music in movie trailers vary according to their respective genres. It is believed in this thesis that the genre labels of movies may be predicted using the information about speech and music presence in their trailers. With this motivation, the next chapter provides a study about the usefulness of speech music information in predicting the genre of movie trailers. The

feature-classifier proposals developed in this thesis for speech and music detection are further utilized for the movie genre identification task.

6

Movie Genre Classification Using Speech-Music Information

Contents

Objective

The previous chapters have primarily focused on the efficient detection of speech and music signals. These signals were present either in isolation or as challenging overlapped mixtures. Encouraging detection performances were observed for audio signals with seamless transition between the categories (speech, music, and speech+music). These experimental results and observations have motivated the application of the proposed techniques to movie audio signals which are predominantly composed of speech and music. Recent approaches have used movie trailers for the task of genre prediction. The movie trailers are designed to concisely represent the whole movie and allude to its key events. Movie editors frequently use specific background music to highlight the mood of different scenes. The speech and music content vary according to movie genres. The information about speech and music signals in a movie trailer audio might relate to its genre label(s). Hence, this chapter performs movie genre classification using the features proposed in the preceding chapters and speech-music prediction probabilities obtained using the models developed in this thesis. An Attention-based Convolutional Neural Network classifier is proposed for the task. The proposal of this chapter is validated on the Moviescope *dataset, and two baseline methods are employed to benchmark the results. The proposed features are observed to perform better than the baselines. Moreover, the current proposal provides acceptable generalization performance over data from a different domain (*EmoGDB*).*

6.1 Task overview

The rise of Over-The-Top (OTT) media platforms has proliferated the creation and consumption of movies and other similar content. Automatic methods for analyzing such rapidly growing content are gradually becoming inexorable. Among various movie analysis use-cases, automatic identification of movie genres may be helpful in underage censorship, targetted publicity, efficient archival, and retrieval. Genre classification of short movie trailers ($\approx$ 3 minutes) has attracted many researchers in the past. Movie trailers are designed in a particular manner so that different emotional responses may be evoked among the viewers [36]. Trailers usually contain rich and varied content representing the theme of the actual movie. Existing approaches have mainly focused on either audio-visual or only visual features for automatic movie genre classification. In comparison, only audio modality is relatively less explored. Accordingly, this chapter proposes approaches to determine probable genres in a movie from its trailer audio. Hence, the task is termed Movie Trailer Genre Classification (MTGC)

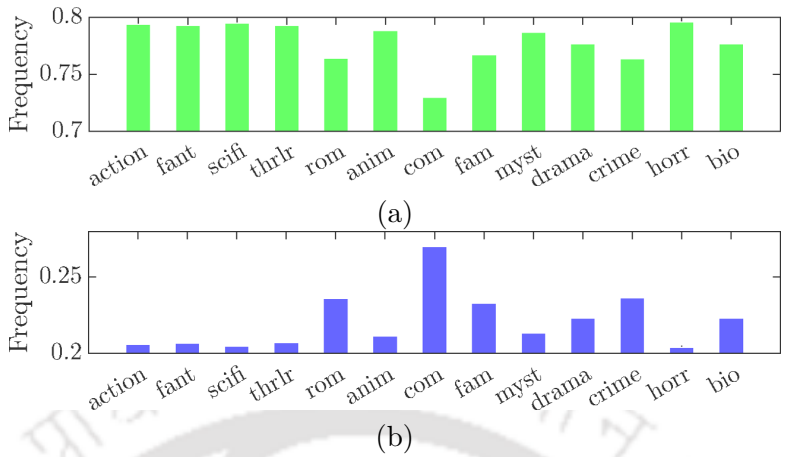


Figure 6.1: Illustrating the genre-wise distribution of music (a) and speech (b) prediction sequences obtained for the trailer audio from the *Moviscope* dataset.

in this chapter.

Existing works have used either the visual modality in isolation [19, 47, 48, 66, 68, 80, 83], or in combination with the audio modality [15,23,27,28,38,41–43,46,65,67,69,70,72,73,75,76,79,83–85,93] for the MTGC task. A few works have used the Bag-of-Visual-Words features [15,47,68]. Encouraging results were also obtained with only the audio modality [24, 45, 49, 92]. The Mel-Frequency Cepstral Coefficients (MFCC) are the most popular audio features. In addition to the audio-visual features, many authors have used additional information from sources like plot summaries, dialogues, posters and movie meta-data in the genre classification task [28,73,75,84,85,93]. The popularly used classifiers in this task are Convolutional Neural Networks (CNN) [72], Support Vector Machines [73, 76, 79], ensemble of Multi-Layer Perceptrons [74], Convolutions Through Time-based residual CNN [20, 75], Recurrent Neural Networks [28,36], Multi-Label K-Nearest Neighbors [84], and unsupervised clustering based methods [85, 92, 201].

Despite the popularity of visual modality in the task of Movie Trailer Genre Classification (MTGC), the audio component has also been found to be useful [75]. The auditory stream is a rich medium for provoking various emotions [38]. Some specific sounds and music are frequently used by movie editors to elicit specific emotional responses and to promote dramatic effects [23]. Audio information has also been found to aid in the better detection of violent scenes [45, 46, 65]. Music used in movies of high-intensity genres like action and horror has very distinct characteristics from those with softer emotional expressions, like drama and romance [24].

Speech, music, and sound effects are the audio types that are frequently found in almost all

movie scenes [97]. Genre-wise distribution of music and speech signals in the *Moviescope* dataset is shown in Fig. 6.1. To the best of our knowledge, previous works in MTGC have performed sound detection using only standard audio features, which might be susceptible to degraded performance in a generic setting. We believe that an improved speech-music detection system might be beneficial for the MTGC task. More confident signal-type predictions will aid in developing better features and classifiers. This work employs the speech-music detection methods proposed in the previous chapters for the current task. Wang et al. [37] noted that there is a semantic gap between human-level interpretations like movie genre and standard audio features. It may not be easy to learn genre-specific information directly from the raw features. Thus, these raw features may be processed further to derive intermediate representations that inherently capture the audio signal-type information. Such latent information might be more beneficial than raw features in mapping the relationship between movie trailer audio and their corresponding genre labels. This chapter proposes the usage of feature representations derived from Gaussian Mixture Models (GMM) proposed in chapter 3 that capture the information of speech and music in the trailer audio. Other hand-crafted features (in chapter 4) and raw representations for automatic-feature learning (in chapter 5) proposed in this thesis are also explored in the MTGC task. Finally, this chapter combines all the various feature representations used in this thesis to obtain an integrated MTGC system. The main contributions of this chapter are summarized next.

(i) The GMM-based CBoW features (see chapter 3), statistical features of spectral peak traces, and speech-music prediction sequences are employed for genre classification of movie trailers. These features encapsulate the information of speech and music present in a movie trailer audio (section 6.2.1).

(ii) The hand-crafted phase-based features described in section 4.2 are employed in the MTGC task.

(iii) Harmonic-percussive decompositions used to automatically learn features discussed in subsection 5.2.1 are also employed in the MTGC task.

(iv) Use of automatic feature learning with harmonic-percussive decompositions in the MTGC task (subsection 5.2.1)

(v) Exploration of attention-based sequence modeling of audio features for the MTGC task (section 6.2.6).

Rest of the chapter is organized as follows. Brief descriptions of the features are provided in subsections 6.2.1, 6.2.2, 6.2.3 and 6.2.5. The proposed classifier is described in subsection 6.2.6. The experimental design and the obtained results are discussed in section 6.3. Finally, the chapter is summarized in section 6.4.

6.2 Proposed approach

This work attempts to segment movie trailer audio into speech and music and use that information for MTGC. Since speech and music are frequently found audio types in movies [97], information about their presence in movie trailers might be related to the movie genre. Chapter 3 of the thesis has proposed a spectral peak tracking-based speech-music classification method. The CBoW features proposed in subsection 3.3.3 encode the information of speech and music present in a given audio signal. Such secondary representations extracted from trained machine-learning models are expected to provide a better mapping between human-level interpretation-based genre labels and audio features. Moreover, various statistical features computed from the peak traces are also used as features. This thesis also explored various hand-crafted features for detecting speech and music. The phase-based features described in chapter 4 are shown to provide improved speech-music detection performance in combination with magnitude-based features. Hence, the phase-based features are also adopted for the MTGC task in this chapter. Furthermore, the harmonic-percussive decompositions used to automatically learn discriminating features in chapter 5 are also explored in the MTGC task. Last but not the least, the speech-music prediction probability sequences of movie trailer audio obtained from the proposals of chapters 3, 4 and 5 are also used as additional features for genre classification. In the upcoming subsections, the feature extraction procedures are briefly described.

6.2.1 Features derived from GMM

Let, $\mathbf{x}[n]$ $(n = 0, \ldots (N-1))$ be a movie trailer audio of N samples. Also, let $X[k,t]$ $(k = 0, \ldots (N_f - 1),\ t = 0, \ldots (n_t - 1))$ be its DFT magnitude spectrogram with N_f frequency-bins and n_t short-term frames of size t_w ms with a shift of t_s ms. For each frame spectra in X, p prominent spectral peaks are identified. The amplitude and frequency information of the selected spectral peaks are retained in two $p \times T$ sized matrices $\mathcal{A}$ and $\mathcal{L}$, respectively. The sequence of r^{th} $(r = 0, \ldots (p-1))$ spectral peak amplitude or location across the audio signals is termed as peak traces. The peak traces capture the distinct striation patterns in the time-frequency representation of speech and music

signals. The distributions of these peak traces are modeled using univariate Gaussian Mixture Models (GMM), trained separately for speech and music signals.

A K-mixture GMM is trained for each r^{th} peak-trace ($r = 0, \ldots (p-1)$) across the training set. Thus, a total of p GMMs are trained separately for peak-amplitude of music (say ${}_m\mathcal{G}_{\mathcal{A}}^{r=0,\ldots(p-1)}$), peak-location of music (say ${}_m\mathcal{G}_{\mathcal{L}}^{r=0,\ldots(p-1)}$), peak-amplitude of speech (say ${}_s\mathcal{G}_{\mathcal{A}}^{r=0,\ldots(p-1)}$) and peak-location of speech (say ${}_s\mathcal{G}_{\mathcal{L}}^{r=0,\ldots(p-1)}$). Subsequently, for every p prominent peak in an audio frame, K posterior probabilities are obtained separately from the speech and music GMMs. Thus, a $2pK$-dimensional feature vector $V_{\mathcal{A}}$ is obtained for each short-term frame by concatenating the posterior probabilities from music and speech peak-amplitude GMMs. Similarly, a $2pm$-dimensional feature vector $V_{\mathcal{L}}$ is obtained by concatenating the posterior probabilities from music and speech peak-location GMMs. The feature vectors $V_{\mathcal{A}}$ and $V_{\mathcal{L}}$ are averaged over 1 s segments to smooth the fluctuations introduced by the possible presence of non-speech and non-music signals in movie trailers. A more detailed description of the feature computation process is provided in subsection 3.3.3.

The proposal in chapter 3 used the *MUSAN* dataset [203] for feature computation. The movie trailer datasets (*Moviscope* and *EmoGDB*) do not provide annotations for speech and music present in the trailer audio. Hence, this work uses GMMs trained using the *MUSAN* dataset to compute the $V_{\mathcal{A}}$ and $V_{\mathcal{L}}$ features for the MTGC task. These features will be collectively referred to as GMM-Posterior Features (GPF) unless mentioned otherwise.

6.2.2 Statistical features

In addition to modeling the peak information distribution using GMMs, this chapter also employs various statistical measures to capture the segment-level statistical information about speech and music in trailer audio. In this context, twelve different statistical measures are computed over the amplitude and location information of peak traces. The various measures computed over the 1 s segments are maximum, minimum, median, mode, mean, standard deviation, geometric mean, geometric standard deviation, harmonic mean, entropy, skewness, and kurtosis. These 12 measures computed from the p peak amplitude traces are concatenated to obtain a $12p$-dimensional feature vector $U_{\mathcal{A}}$. Similarly, a $12p$-dimensional feature vector $U_{\mathcal{L}}$ is obtained from the p peak location traces. Unless mentioned otherwise, these features will be collectively referred to as Statistical-measure Features (SF).

6.2.3 Hand-crafted features

Chapter 4 describes three phase-based features, viz. HNGDCC (see subsection 4.2.1), MGDCC (see subsection 4.2.2) and IFCC (see subsection 4.2.3). These features capture the phase characteristics of the underlying signal. It has been shown in chapter 4 that the phase-based features, in combination with the magnitude-based features, improve speech-music classification performance. Therefore, it may be inferred that the phase component of speech and music signals carries complementary information to the magnitude component. Since speech and music are the dominant audio types in movie trailers, the phase component might capture some genre-specific additional information that might complement the magnitude-based features. Hence, the three 39-dimensional phase-based features are explored in this chapter for the MTGC task. These features will be referred to as Phase-Based Features (PBF) unless mentioned otherwise.

6.2.4 Raw representations

The harmonic percussive source separation (HPSS) based features (see subsection 5.2.1) used in chapter 5 provided significant improvement to speech+music detection compared to the baseline methods. It is known that a considerable portion of movie trailer audio is composed of speech+music signals. Features that can detect speech+music signals efficiently might also aid in the MTGC task. Hence, this chapter adopts the HPSS-based Log-Mel Harmonic Spectrogram (LMHS) and Log-Mel Percussive Spectrogram (LMPS) based features for the MTGC task. The LMHS and LMPS are computed with $l_{\text{harm}} = 21$, $l_{\text{perc}} = 11$ and $n_{\text{mel}} = 128$ (see subsection 5.2.1). These features will be referred to as Harmonic Percussive Features (HPF) unless mentioned otherwise.

6.2.5 Speech-music prediction feature

In addition to the GMM-Posterior features (GPF), statistical features (SF), hand-crafted, and raw features, this chapter also utilizes the speech-music prediction sequences for movie trailer audio as a feature. The Speech vs. Music Prediction probabilities (SMP) sequence for a movie trailer audio is computed for consecutive 1 s segments of the audio. The classifiers trained on *MUSAN* dataset in chapters 3, 4 and 5 are used to obtain the SMP. It may be noted that separate MTGC systems are developed by using the features proposed in each previous chapter. The SMP obtained using the trained models from each previous chapter is used with their respective MTGC systems. A median filter of kernel width 5 (samples) is used to suppress the prediction noise in SMP. The sequence

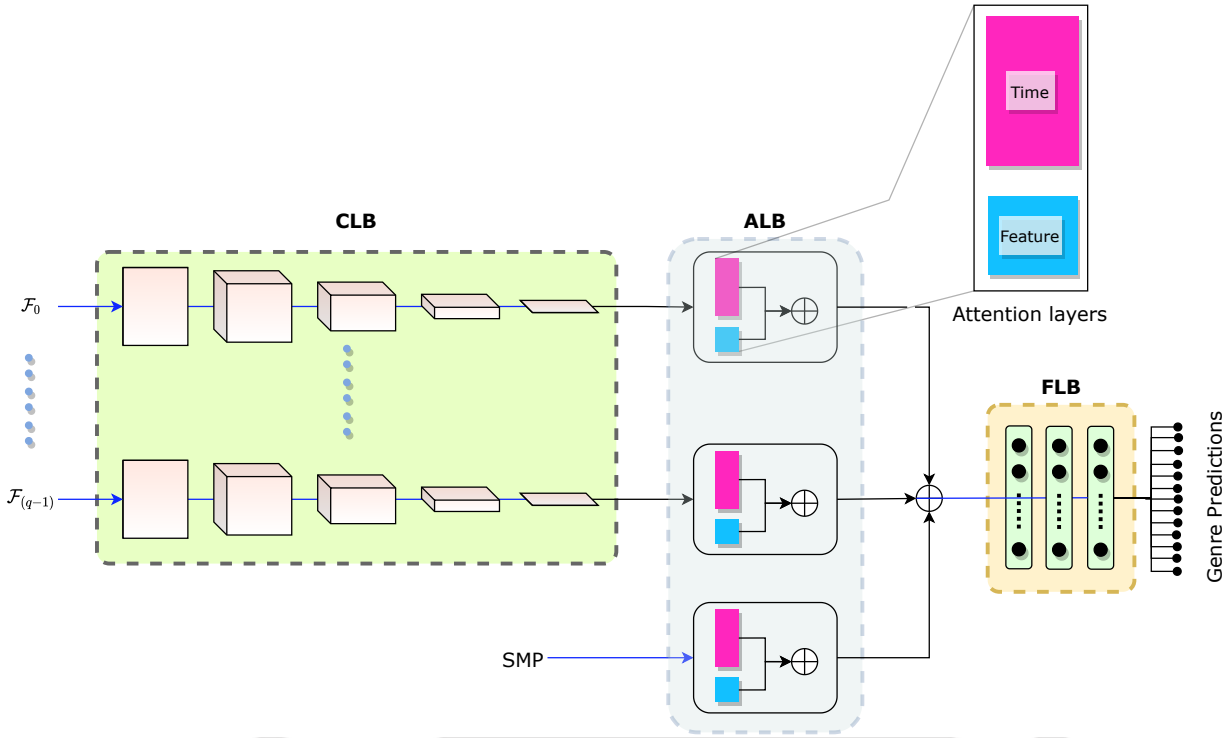


Figure 6.2: Proposed Attention-based deep CNN classifier for MTGC.

of smoothed SMP is directly passed through attention layers (see subsection 6.2.6) to model their relationship with the genre of a movie trailer. The classifier architecture proposed in this chapter is described in the following subsection.

6.2.6 Classifier architecture

This chapter proposes an Attention-based Convolutional Neural Network (ACNN) classifier for the MTGC task. To the best of our knowledge, the Attention mechanism has not been previously explored for aggregating audio features in the MTGC task. A block diagram of the proposed architecture is shown in Fig. 6.2. The ACNN consists of three distinct blocks added in sequence, viz., Convolutional Layer Block (CLB), Attention Layer Block (ALB), and Fully-connected Layer Block (FLB). The ACNN can be used with single feature input ($\mathcal{F}_0$) or q multiple feature inputs ($\mathcal{F}_0, \ldots \mathcal{F}_{(q-1)}$). Separate CLB-ALB units can be added as parallel branches when using ACNN as a multiple-input system. The CLB is equipped with *Maxpooling* layers to reduce the feature dimension gradually to unity. The time axis of the input features is not pooled in the CLB. The convolutional layers learn to encode the required discriminating information along the filter (or channel) axis. The number of convolutional filters in the CLB is experimentally optimized. The outputs of all convolutional layers

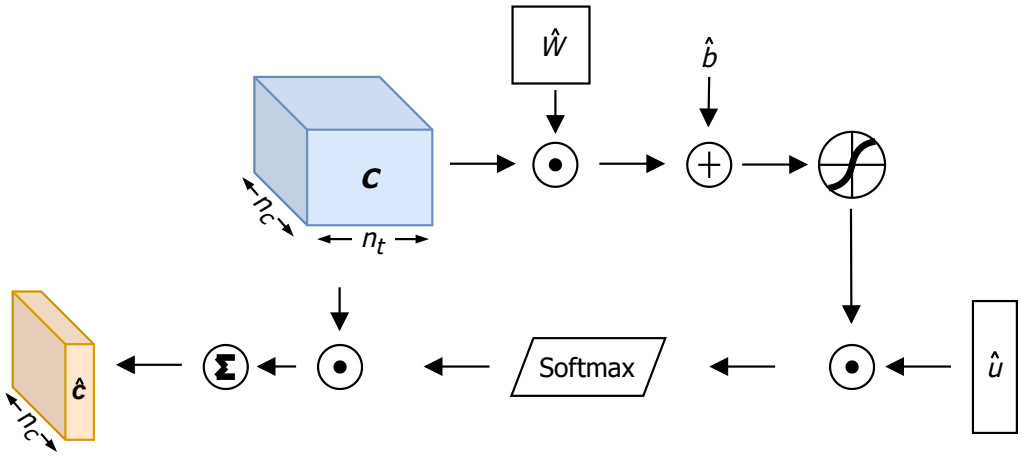


Figure 6.3: Block diagram of the attention module for computing $\hat{c}$.

are passed through *Linear* activation and *Batch Normalization*. Except for the SMP features (see subsection 6.2.5), all other features (described in subsection 6.2.1, 6.2.2 and 6.2.3) are fed to the CLB units. The 2-dimensional SMP features are directly fed to an ALB unit (see Fig.6.2). The attention mechanism used in the ALB units is described next.

The attention mechanism used in this work is motivated by the one proposed in [298]. This chapter employs attention to collate representations learned by the convolution layers along the time and channel axes (see Fig.6.3). The time-axis attention emphasizes time steps that are important for the underlying task. On the other hand, channel-axis attention aims to capture the most informative filters in the last convolutional layer. Let $\mathbf{C}[l, s]$ be a $n_t \times n_c$ representation obtained from the last convolutional layer, where $l = 0, \ldots (n_t - 1)$ and $s = 0, \ldots (n_c - 1)$ represent the time and channel axes, respectively. To perform time-axis attention, $\mathbf{C}[l, s]$ is fed through a multi-layer perceptron with a $n_t \times n_t$ weight matrix $\hat{W}$ and a $n_t \times 1$ bias vector $\hat{b}$ to obtain a $n_t \times n_c$ hidden representation $\hat{h}$ (equation 6.1).

$$\hat{h} = \tanh\left(\hat{W}\mathbf{C} + \hat{b}\right) \tag{6.1}$$

Next, a trainable $n_t \times 1$ weight vector $\hat{u}$ is used to obtain context-weights $\hat{\alpha}$ for every time step (equation 6.2).

$$\hat{\alpha}_{n_t \times 1} = \text{softmax}\left(\hat{h}^T \hat{u}\right) \tag{6.2}$$

Finally, a n_c-dimensional time-axis attention-weighted context-vector $\hat{\mathbf{c}}$ is obtained (equation 6.3).

$$\hat{\mathbf{c}} = \sum_{l=0}^{(n_t-1)} \hat{\alpha}[l] \cdot \mathbf{C}[l,:] \tag{6.3}$$

Proceeding in a similar fashion, a n_c-dimensional channel-axis attention-weighted context-vector $\tilde{\mathbf{c}}$ is obtained using another attention layer represented by $\left\{\tilde{W}, \tilde{b}, \tilde{u}\right\}$. At last, the context vectors are concatenated to form a $(n_t + n_c)$-dimensional vector $\mathbf{c} = [\hat{\mathbf{c}}, \tilde{\mathbf{c}}]$ and fed forward for classification. Basically, the ALB units perform a weighted sum along the time and channel axes to project the three-dimensional CLB output to one-dimensional vectors. The outputs obtained from the ALB of each feature branch are concatenated and passed to the FLB. This feature combination strategy is known as Intermediate Fusion (IF). The FLB unit is described next.

The ACNN has only one FLB. The time and channel attention output from the ALB of each feature branch are concatenated and fed through a series of three 300-neuron fully-connected layers in the FLB. The outputs obtained from the FLB are passed to a *Sigmoid* activated output layer that predicts the probability of each genre. The number of nodes in the prediction layer equals the number of unique genre labels in the data. For the *Moviescope* dataset, 13 output nodes are added to the prediction layer. The output nodes have a *Sigmoid* activation and are trained with a *Binary Crossentropy* loss function. Network optimization is performed using the *Adam* optimizer [231] with an initial learning rate of 0.001. Each of the fully-connected layers are followed by *Batch Normalization*, *ReLU* activation and a *Dropout* factor of 0.1. The hyperparameters of ACNN have been finalized after performing a grid search over various possible values. The next section discusses the experiments performed and their results.

6.3 Experiment and results

Recently, Cascante et al. [28] published a multi-modal movie trailers dataset called *Moviescope* that consists of approximately 5000 trailer videos, plot summaries, posters, and various other metadata. Movies in the dataset are labeled with one or more genre labels from a list of 13 possible genres. Since the *Moviescope* dataset is one of the most extensive datasets available, the proposed MTGC method has been benchmarked on this dataset. Only the audio component from the trailer videos is extracted and used in this work. The features developed in the previous chapters of this thesis are explored in this chapter to perform the MTGC task. The audio signals are processed with a sampling

rate $f_s = 16000$Hz. The GPF and SF (see subsection 6.2.1 and 6.2.2) are computed with short-term window size of $t_w = 10$ ms with a frame size of $t_s = 5$ ms. The PBF and HPF (see subsection 6.2.3) are computed with short-term window size of $t_w = 25$ ms with a frame size of $t_s = 10$ ms. The Hamming window is applied to the short-term frames to suppress windowing effects. The spectrograms have been computed with $N_f = 10^{-3} \times t_w \times f_s$ frequency bins. Spectral peak tracking is performed with $p = 10$ prominent peaks, and peak-trace GMMs are trained with $m = 5$ mixtures. The codes used in this work are shared publicly [1].

A 30 s segment is used as input to the system, following the proposal of Cascante et al. [28]. The features $V_{\mathcal{A}}$ and $V_{\mathcal{L}}$ represent a 1 s segment. Thus, both the $V_{\mathcal{A}}$ and $V_{\mathcal{L}}$ features are 2-dimensional matrices of size 30×100. Mini-batches of these feature vectors are provided as input to ACNN classifier (see Fig. 6.2). The statistical features $U_{\mathcal{A}}$ and $U_{\mathcal{L}}$ are presented to the classifier with an input size of 30×120 each. The PBF features are provided as input to the ACNN classifiers in the form of 39×3000 feature matrices. The HPF features are input as 128×3000 sized feature matrices to the ACNN classifier. The speech-music probabilities for the movie trailer audio are predicted for 1 s segments with a shift of 1 s. Thus, the SMP feature is fed as a 2×30 feature matrix to represent a 30 s interval. The pooling design in the CLB units for each feature is modified accordingly to fulfill the minimum structural requirements of the ACNN classifier, as described in subsection 6.2.6.

The proposed approach involves the use of multiple features for the classification task. In this regard, two different feature fusion strategies have been used in this chapter. First, the intermediate fusion (IF) strategy is described in subsection 6.2.6. Second, a Late Fusion (LF) strategy is also explored to combine individually trained models. A weighted sum of the genre-wise logits predicted by separate models is used as the final prediction score in an LF strategy. The optimal weights for different models are determined experimentally. It may be noted that the results of feature combinations are reported as "$S_X := F_1 + \ldots + F_N$ (LF/IF)". This signifies that a system "S_X" is formed with a combination of features "$F_1 \ldots F_N$" combined either through late-fusion (LF) or intermediate-fusion (IF). Performances in this work are reported as the area under precision-recall curves ($AU(\overline{PRC})$) for each genre separately and also their macro, micro, and sample averages [28].

[1] Available online at: `https://github.com/mrinmoy-iitg/MTGC_Speech_Music_Segmentation`

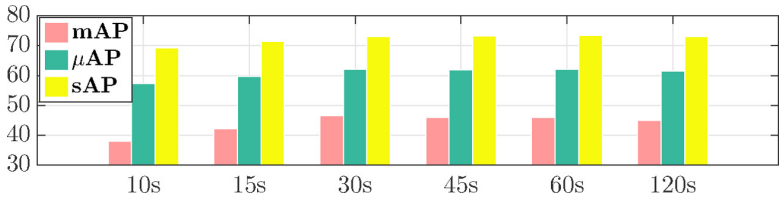


Figure 6.4: Illustrating the performance variation of MTGC systems based on the GPF+SF+SMP (IF) feature combination across different segment durations.

6.3.1 Baseline methods

The performances of the proposed genre classification methods are compared with two recent audio-based approaches from the literature. The first baseline $B1$ uses the proposal of Sharma et al. [92]. This method ($B1$) involves a set of 68 standard tempo-spectral audio features and K-Means clustering-based audio segmentation information. Second, the creators of *Moviescope* dataset [28] provided a baseline performance. The authors proposed the use of Log-Mel Spectrograms (LMS) of 30 s segments as input to a Convolutional Recurrent Neural Network classifier for the MTGC task. Performance comparisons in this work are made with only the audio-based results ($B2$) of Cascante et al. [28]. The baseline performance is directly quoted from the paper. The proposed features are combined with the LMS feature in later experiments to obtain improved performances. The LMS feature is computed with the same settings as the baseline $B2$. Cascante et al. [28] computed 2048-point DFT spectrograms with a short-term frameshift of 256 samples for audio signals sampled at 12000kHz. They computed the LMS feature with 128 Mel filters. Thus, the LMS feature is provided as a 128×1407 matrix (see [28] for details) to a CLB-ALB unit of the ACNN classifier.

6.3.2 Performance analysis

The effect of different segment durations on the classification performance of the combination of GPF+SF+SMP is shown in Fig. 6.4. With the increase in segment size, there is a general trend of performance improvement. A significant improvement is observed when segment size is increased to 30 s. After that, a saturation of the performance is observed. Thus, 30 s segment durations appear to be optimal for training a classifier on the *Moviescope* dataset.

Performance of the proposed features in MTGC with ACNN classifier are presented in Tables 6.1, 6.2, 6.3, and 6.4. Table 6.1 lists the performance of the spectral peak tracking based features proposed in chapter 3. It can be observed that the intermediate-fusion of GPF, SF, and SMP features ($P4$ in

Table 6.1: Performances of baseline and proposed spectral peak tracking based features in the MTGC task. Here, $P1$:=GPF, $P2$:=GPF+SF (IF), $P3$:=GPF+SMP (IF), $P4$:=GPF+SF+SMP (IF), $P5$:=GPF+SF+SMP+LMS (IF), and $P6$:=GPF+SF+SMP+LMS (LF). Performances better than the baselines are highlighted in **bold-face**, while the best performance among the proposed features is underlined.

Feature	action	anim	bio	com	crime	drama	fam	fant	horr	myst	rom	scifi	thrlr	mAP	μAP	sAP
$B1$	43.77	20.83	8.53	76.79	27.91	72.07	26.71	14.86	24.18	20.39	38.55	24.85	46.80	34.74	53.17	68.47
$B2$	56.70	48.00	11.20	86.20	40.00	79.00	49.60	44.70	37.60	22.70	43.00	27.00	56.30	46.30	61.40	72.30
$P1$	61.31	42.34	10.19	85.85	38.58	78.34	43.50	22.19	42.11	29.74	41.19	25.18	57.96	44.83	59.81	71.23
$P2$	58.62	42.76	10.70	86.19	37.25	79.95	41.94	26.61	41.18	29.37	38.88	25.92	60.16	44.92	62.00	73.35
$P3$	61.68	43.26	9.70	85.67	38.84	78.49	42.99	23.94	42.25	31.39	42.21	25.27	58.49	45.26	59.77	71.06
$P4$	58.85	51.14	**11.74**	85.90	37.52	79.87	46.55	27.35	40.99	**32.92**	41.53	**27.37**	60.65	46.62	62.03	73.15
$P5$	**63.53**	54.62	11.70	**87.43**	38.89	**81.53**	53.98	35.13	48.75	29.30	42.97	27.14	**62.39**	49.34	64.90	75.53
$P6$	37.05	**67.40**	9.86	86.65	**40.79**	78.83	**61.74**	37.05	**50.28**	24.55	42.17	22.87	60.14	**51.48**	**65.57**	**75.78**

Table 6.1) with the ACNN classifier performs better than both the baselines $B1$ and $B2$. The proposed system performs poorly for the comedy, crime, family, fantasy, and romance genres but performs better for others. However, the system performance improves significantly when GPF, SF, and SMP features are combined with a raw feature like LMS. In an intermediate fusion with the LMS feature ($P5$ in Table 6.1), the system provides lower performance only for the crime, fantasy, and romance genres. However, the macro, micro, and sample average values improve significantly. In a LF setting ($P6$ in Table 6.1), the average performances improve even further. The $P6$ system significantly improves the detection of animation, family, horror, and thriller genres. It may be noted that GMMs trained on the *MUSAN* dataset are used in this work. The performance of the proposed features might improve if actual speech-music annotation for the movie trailer audio is available.

Table 6.2 lists the performance of PBF in the MTGC task. The individual performances of the three PBFs are not satisfactory enough. The HNGDCC is the only PBF that provides slightly comparable performance to $B1$ ($P1$ in Table 6.2). The IF of HNGDCC, MGDCC and SMP ($P4$ in Table 6.2) happens to be the best PBF combination. Such results are a reiteration of the observation of chapter 4 where the individual performance of phase-based features was lower than magnitude-based features. The principles of movie editing involve synthetic stylistic mixing of various non-similar sounds. Such processes might also corrupt the phase relationships of the component signals. Nevertheless, the phase-based features were found to improve the speech-music classification performance in combination with the magnitude-based features in chapter 4. A similar effect is observed for the MTGC experiment as

Table 6.2: Performances of baseline and phase-based features in the MTGC task. Here, $P1$:=HNGDCC+SMP (IF), $P2$:=MGDCC+SMP (IF), $P3$:=IFCC+SMP (IF), $P4$:=HNGDCC+MGDCC+SMP (IF), $P5$:=HNGDCC+IFCC+SMP (IF), $P6$:=MGDCC+IFCC+SMP (IF), $P7$:=HNGDCC+MGDCC+IFCC+SMP (IF), $P8$:=HNGDCC+MGDCC+IFCC+SMP+LMS (IF). The performances better than the baselines are highlighted in **bold-face**, while the best performance among the proposed features is underlined.

Feature	action	anim	bio	com	crime	drama	fam	fant	horr	myst	rom	scifi	thrlr	mAP	μAP	sAP
$B1$	43.77	20.83	8.53	76.79	27.91	72.07	26.71	14.86	24.18	20.39	38.55	24.85	46.80	34.74	53.17	68.47
$B2$	56.70	48.00	11.20	86.20	40.00	79.00	49.60	44.70	37.60	22.70	43.00	27.00	56.30	46.30	61.40	72.30
$P1$	49.44	29.90	10.92	74.81	21.51	70.39	31.88	18.93	20.43	20.15	34.62	24.22	46.32	35.45	51.64	66.43
$P2$	38.72	8.30	7.63	68.15	24.00	61.27	16.10	12.65	17.87	14.48	32.80	19.78	38.26	28.02	45.86	63.82
$P3$	40.81	15.71	7.86	71.14	24.33	67.69	23.26	14.06	16.80	19.12	30.22	22.36	44.28	31.01	49.03	64.88
$P4$	49.58	16.18	10.65	76.74	21.44	68.86	28.57	19.33	24.67	19.33	36.30	24.12	45.67	34.42	52.20	67.36
$P5$	42.20	14.67	8.11	69.13	23.52	67.67	24.20	13.13	16.95	19.58	29.89	22.75	45.28	30.95	48.56	64.74
$P6$	40.70	12.72	7.07	69.37	24.74	67.92	22.20	13.15	16.87	18.52	30.16	22.58	42.79	30.28	48.74	64.69
$P7$	42.90	10.61	6.72	71.36	26.44	67.04	21.88	12.93	16.99	20.85	30.45	21.24	46.33	30.79	48.62	64.68
$P8$	**64.60**	45.72	10.56	**87.72**	37.54	**81.45**	**50.80**	27.94	36.59	**24.42**	42.16	26.95	**57.52**	46.02	**63.31**	**73.94**

well. The combination of PBF, SMP, and LMS ($P8$ in Table 6.2) provides the best micro and sample averaged performance. This combination improves the performance of action, comedy, drama, family, mystery, and thriller genres. Apart from the comedy genre, the remaining genres are music-intensive (see Fig. 6.1). Also, these genres are known to consist of many sound effects. From experimental observations, it may be concluded that the phase information helps in the genre identification of movie trailers with high amounts of music and sound effects.

The performance of HPF in MTGC is presented in Table 6.3. The combination of LMHS and LMPS features ($P3$ in Table 6.3) provides better performance than $B1$. The addition of SMP feature further improves the performance (see $P4$ in Table 6.3). The SMP features in this experiment also include speech+music probabilities, which might explain the improvement obtained by adding it. The intermediate fusion of LMHS, SMP, and LMS provides the best sample average performance and the best AUC for action, animation, comedy, and drama genres. However, the HPFs are unable to outperform the baseline throughout. A possible reason for such a performance might be the smoothing step involved in the harmonic-percussive decomposition algorithm. Such smoothing might not have affected the relatively easier 3-class single-label classification task of chapter 5. However, the information smoothing might have been detrimental to the 13-class multi-label classification problem in the case of MTGC. Nonetheless, the HPF provides comparable performance and may be explored in detail in the future.

Table 6.3: Performances of baseline and harmonic-percussive features in the MTGC task. Here, $P1$:=LMHS, $P2$:=LMPS, $P3$:=LMHS+LMPS (IF), $P4$:=LMHS+LMPS+SMP (IF), $P5$:=LMHS+LMPS+LMS (IF), $P6$:=LMHS+LMPS+SMP+LMS (IF), $P7$:=LMHS+SMP+LMS (IF). The performances better than the baselines are highlighted in **bold-face**, while the best performance among the proposed features is underlined.

Feature	action	anim	bio	com	crime	drama	fam	fant	horr	myst	rom	scifi	thrlr	mAP	μAP	sAP
$B1$	43.77	20.83	8.53	76.79	27.91	72.07	26.71	14.86	24.18	20.39	38.55	24.85	46.80	34.74	53.17	68.47
$B2$	56.70	48.00	11.20	86.20	40.00	79.00	49.60	44.70	37.60	22.70	43.00	27.00	56.30	46.30	61.40	72.30
$P1$	49.70	20.55	10.13	83.06	28.73	73.27	24.80	13.31	26.26	22.62	38.04	21.75	54.04	36.34	53.53	67.72
$P2$	40.34	10.14	7.42	81.26	24.30	59.07	19.56	11.76	22.73	21.24	34.90	20.48	47.11	31.19	49.27	65.74
$P3$	56.12	15.31	7.88	85.05	25.44	75.65	28.12	15.21	26.83	21.03	38.20	22.86	51.62	36.48	56.34	69.57
$P4$	52.08	33.84	8.37	84.12	31.13	76.39	37.40	18.47	34.31	**26.20**	37.86	25.11	52.57	40.19	57.87	71.03
$P5$	57.28	47.50	8.70	85.64	30.50	77.93	44.14	23.44	34.89	24.49	38.87	24.19	52.77	42.68	59.99	72.45
$P6$	57.91	45.97	8.60	85.68	30.94	78.80	44.95	26.19	34.82	24.93	38.57	24.34	52.64	42.97	60.19	72.52
$P7$	**58.04**	**52.75**	9.11	**86.33**	29.79	**79.51**	48.51	28.55	36.27	23.92	38.96	25.40	54.64	44.35	61.12	**73.19**

Table 6.4: The combined performances of the three proposals from chapters 3, 4 and 5 in the MTGC task. Here, $S1$:=GPF+SF+SMP (IF), $S2$:=HNGDCC+MGDCC+SMP (IF), $S3$:=LMHS+LMPS+SMP (IF), $S4$:=$S1$+$S2$+$S3$ (LF). The Performances better than the baselines are highlighted in **bold-face**, while the best performance among the proposed features is underlined.

Feature	action	anim	bio	com	crime	drama	fam	fant	horr	myst	rom	scifi	thrlr	mAP	μAP	sAP
$B1$	43.77	20.83	8.53	76.79	27.91	72.07	26.71	14.86	24.18	20.39	38.55	24.85	46.80	34.74	53.17	68.47
$B2$	56.70	48.00	11.20	86.20	40.00	79.00	49.60	44.70	37.60	22.70	43.00	27.00	56.30	46.30	61.40	72.30
$S1$	**58.85**	51.14	**11.74**	85.90	37.52	79.87	46.55	27.35	40.99	**32.92**	41.53	27.37	60.65	46.62	62.03	73.15
$S2$	49.58	16.18	10.65	76.74	21.44	68.86	28.57	19.33	24.67	19.33	36.30	24.12	45.67	34.42	52.20	67.36
$S3$	52.08	33.84	8.37	84.12	31.13	76.39	37.40	18.47	34.31	26.20	37.86	25.11	52.57	40.19	57.87	71.03
$S4$	27.83	**53.29**	11.51	**86.49**	38.48	**80.37**	48.13	27.83	**42.18**	32.74	41.35	**27.64**	**60.70**	**47.01**	**62.40**	**73.29**

Finally, all the proposed systems developed for MTGC in the previous subsections are combined through late fusion (LF). This combination helps in determining if these systems capture any complementary information to benefit the underlying task. Table 6.4 lists the performances of combining all the previous systems. The combination of GPF, SF and SMP is referred to as $S1$ in Table 6.4. The combination of HNGDCC, MGDCC, and SMP is indicated by $S2$. The system $S3$ indicates the combination of LMHS, LMPS, and SMP. All the systems considered in this experiment consist of only the features developed in this thesis. The combined system ($S4$ in Table 6.4) provides the best performance for animation, comedy, drama, horror, sci-fi, and thriller genres. The combined system also provides the best macro, micro, and sample averaged performance. Thus, it may be inferred that the features proposed in this thesis are also efficient in the MTGC task. The system $S1$

Table 6.5: Generalization performance on *EmoGDB* dataset. Here, $P1$:=GPF+SF+SMP (IF) and $P2$:=$P1$+LMS (LF).

$B2$			$P1$			$P2$		
mAP	**μAP**	**sAP**	**mAP**	**μAP**	**sAP**	**mAP**	**μAP**	**sAP**
48.33	39.04	64.08	45.47	39.85	56.37	50.21	44.21	65.17

individually outperforms the baselines. Such a result indicates that secondary information obtained from the trained GMMs is helpful in a challenging task like MTGC. Moreover, the improvement in performance obtained by the fusion of GMM-based (GPF), statistical (SF), hand-crafted (PBF), and raw representations (HPF) based systems indicate that such a combination can be useful in building an effective MTGC system.

6.3.3 Generalization performance

The generalization performance of the trained MTGC model on a different dataset is also evaluated to establish the viability of the proposed method in MTGC. The best MTGC system obtained so far (GPF+SF+SMP) is utilized in this experiment. For this task, the *EmoGDB* [36] dataset is utilized. The *EmoGDB* dataset consists of 100 Indian movie trailers with a single genre label for each movie. The *Moviescope* dataset predominantly consists of Hollywood movies. Hence, *EmoGDB* can be considered an ideal choice for evaluating the generalization performance of models trained on the *Moviescope* dataset. For this experiment, the model predictions for only the six possible labels from the *EmoGDB* dataset are considered. The genre-wise predictions of both baseline and proposed methods for *EmoGDB* dataset are scaled to $[0, 1]$ over all samples before computing the performance metrics to account for the differences in train and test data. The cross-dataset results of the baseline $B2$ [28] and the proposed methods are presented in Table 6.5. It can be observed that the proposed features alone (see $P1$ in Table 6.5) do not perform better than the baseline in this case. However, the late fusion of proposed features with LMS (see $P2$ in Table 6.5) provides significant improvement. The obtained results suggest that the proposed model can generalize well over an unseen dataset from a different domain.

6.4 Summary

This chapter proposes the use of speech-music prediction probability sequences for the task of movie trailer genre classification. The audio-type cues are encoded in a GMM-based feature computed using spectral peak tracking of audio spectrograms. The GMM-based feature, peak trace statistical measures, and sequence of speech-music probability for trailer audio are explored as features. Moreover, hand-crafted phase-based and raw representations of harmonic-percussive decompositions developed in previous chapters of the thesis are also explored. An Attention-based CNN classifier is used to perform the genre prediction. The results obtained with the proposed approach justify the utilization of speech-music segmentation in movie genre classification. Moreover, the generalization performance of the proposed approach is also found to be satisfactory.

7

Conclusions

Contents

Objective

This chapter summarizes the thesis by highlighting the significant contributions and acknowledging the primary outcomes of the work. In addition, a discussion is provided on possible directions of extension of this work in the future.

7.1 Summary

This thesis attempts to predict the genre of movies using information from its audio. Previous works in movie genre prediction have mainly focussed on the visual modality for the task. The audio was mainly used as an additional informational channel. However, the richness of the movie audio component motivates a detailed study. Therefore, this thesis analyzes the movie audio in considerable detail and proposes novel methods for genre prediction. Rather than using whole movies, recent works have used short trailers for the genre prediction task. Trailers concisely represent the main events of a movie. Movie trailer audio is mainly composed of speech and music signals. Speech in the form of dialogues carries forward the narrative of the movie. Music is used to dramatize the scenes. The significance of these audio types in the movie inspired a detailed study of speech and music signals in this thesis. This thesis proposes novel features and classification schemes for detecting speech and music signals in isolated and overlapping scenarios. The proposed speech and music detection methods are subsequently applied to the task of movie trailer genre prediction. The results obtained validate the hypothesis that speech and music signals in movie audio carry genre-specific information. The main contributions of this thesis are briefly summarized next.

- There are multiple speech-music datasets previously used in the literature. However, the speech and music signals in the available datasets were collected from various sources which were mostly not related to movies. The current thesis aimed to perform SMC in movie audio. Thus, the developed systems must be tested on speech and music signals from movie audio. The lack of such a dataset required that the authors create one. Therefore, a (*Movie-MUSNOMIX*) dataset of 8 hours and 20 minutes consisting of manually annotated audio signals into seven audio classes was created. The audio data was obtained from four Indian movies. Two speech-domain expert annotators labeled the signals. The dataset is shared publicly to foster research using the Indian movie audio signals.

- Chapter 3 proposes two sets of magnitude spectrum based tempo-spectral features for Speech vs.

Music Classification (SMC). The proposed features are computed using this thesis's proposed Spectral Peak Tracking (SPT) method. The spectral peak trajectories obtained using the SPT method are further processed to compute two sets of novel features. The proposed features capture information about distinct striation patterns of speech and music spectrograms. The SPT-based features perform better than many standard feature sets previously used in SMC under various challenging scenarios. The proposed features also provided decent performance in segmenting continuous audio sequences consisting of speech and music signals.

- Chapter 4 analyzes the phase information of speech and music signals. This was previously under-explored in the SMC task. Three existing phase-based audio features originally proposed for tasks like phoneme recognition and speaker recognition are explored in this thesis for the SMC task. It is observed that the phase-based and magnitude-based features have complementary information. The phase-based features have exhibited decent performance in the SMC task. The phase features combined with magnitude features performed better than the baselines considered. Phase-based features combined with magnitude-based ones also provided decent segmentation performance.

- Chapter 5 explores the comparatively challenging case of detecting speech overlapped with music against clean speech or music signals. This thesis proposes the use of Harmonic Percussive Source Separation (HPSS) to generate better features suited for the task. Also, the Multi-Task Learning (MTL) framework is explored to design better classifiers. The HPSS-based features are shown to perform better than the baseline features. Moreover, the MTL-framework based classifiers performed better than the Single-Task Learning-based classifiers. Also, HPSS-based features and MTL-based classifiers are found to be better at segmenting continuous audio sequences of speech and music.

- Chapter 6 proposes the use of speech and music presence information for Movie Trailer Genre Classification (MTGC). The speech and music detection methods developed in this thesis are employed for the MTGC task. The proposals of chapters 3 ($S1$), 4 ($S2$) and 5 ($S3$) are used to develop separate MTGC systems. An Attention-based Convolutional Neural Network (ACNN) is also proposed for the MTGC task. The $S1$ system developed using the proposal of chapter 3 is the best performer and improves upon the results of the baselines. This system also displays

decent generalization performance when tested on a different dataset. The performance is found to improve further when the three proposed systems are combined through a late-fusion strategy. Hence, each of the three systems ($S1$, $S2$, and $S3$) capture some non-overlapping aspects of the underlying task that can be constructively combined for efficient MTGC performance.

7.2 Conclusions of the work

This thesis primarily aimed at using only the audio modality to propose an efficient movie genre prediction system. Thus, the most prominent components of movie audio signals (speech and music) were studied with considerable attention. The results obtained during the thesis work mostly concur with the hypothesis of the research. The different takeaways from this thesis are discussed here.

- **Establishing the importance of hand-crafted features** – The current trend in signal processing research indicates that there is a significant thrust toward large data-driven deep networks. These networks are generally exceedingly successful in solving a vast range of research problems, subject to optimal network architectures and the availability of enough data. Such a trend has inadvertently shifted the focus of the researchers from using hand-crafted features to automatic feature learning-based methods. However, we believe that a healthy combination of domain knowledge and data-driven learning must be the way forward. This belief is also endorsed by the results obtained in chapter 3. The proposed hand-crafted features (chapter 3) are domain knowledge based enhancements over basic features like spectrograms. The proposed features performed very well with DL methods. However, some automatic feature learning based deep-learning methods outperformed the hand-crafted features. Nevertheless, the best result was obtained when the proposed features were combined with the deep-learning-based methods. It is observed that deep-learning based methods are very effective in learning discriminative features. However, such methods are constrained by the amount of discriminative information in the training data. Also, out-of-domain data pose difficulties to such data-driven methods. Therefore, combining automatic feature learning and hand-crafted features might help develop more robust systems.

- **Establishing the importance of phase information** – The application of phase information in the SMC task remains under-explored in the existing literature. Phase is a vital component of audio signals like speech and music. Chapter 4 provides a detailed study of the phase component

of speech and music signals for their classification. The results obtained validate the importance of phase information. Phase alone cannot provide the best discrimination of speech and music signals. The popularity of magnitude-based features is evident in their superior performance individually. Nevertheless, phase-based and magnitude-based features in combination provide better performance than the baselines. This observation implies that the phase captures complementary information that can be utilized with magnitude information to build robust SMC systems.

- **Effectiveness of separating harmonic and percussive components** – Previous works in speech overlapped with music detection have relied on standard audio features for the task. Speech and music can be discriminated by their distinct harmonic and percussive properties. Feature representations that are derived from standard DFT spectrograms include the combined information of harmonic and percussive components. Such representations might not be optimal for the task. This thesis shows that separating the harmonic and percussive components helps in computing more discriminative features. The harmonic-percussive features are shown to improve the detection of speech and music signals in isolated or overlapped conditions. Such features enable the classifier models to focus on one type of striation information at a time and learn better discrimination between the audio categories. In other words, the decomposition of harmonic and percussive components effectively generates two different representations of the same spectrogram. In each representation, one type of pattern is retained while the other is suppressed. This separation of information aids in the classification task.

- **Relation of speech and music signals to the movie genre** – This thesis hypothesized that the information of speech and music in movie trailer audio might relate to its genre. For this purpose, efficient features for speech and music detection were proposed in this thesis. The proposed features and Speech Music Prediction (SMP) confidences were used to predict the movie trailer genres. The proposed approach provided improved performance compared to the baselines. Also, the proposed approach provided decent generalization performance on a different dataset. Such results validate the hypothesis of this thesis. Therefore, it may be reasoned that this thesis establishes the importance of the speech and music signals in movie genre prediction.

7.3 Future extensions

This section discusses possible directions for future extensions of the work presented in this thesis.

- **Extension of the *Movie-MUSNOMIX* dataset:** The *Movie-MUSNOMIX* dataset is created using the audio of just four Indian movies. Even though the movie release dates span a wide range (1966-2011), this dataset cannot represent all the diversities of Indian movie audio. The *Movie-MUSNOMIX* dataset in its current form can only provide limited exposure to the Indian movie soundscape. There is a need to substantially extend the *Movie-MUSNOMIX* dataset so that a detailed study of Indian movies might be possible. Therefore, additional audio annotation of many more Indian movies needs to be added to the *Movie-MUSNOMIX* dataset in the future.

- **Refinement of speech, music and speech+music segments:** The speech, music, and speech+music segments detected using the proposed methods include a lot of other sounds and acoustic effects generally encountered in movies. Such sounds are sparsely distributed across movies and are thus difficult to detect as separate homogenous segments. A possible future extension of the current proposal would be to refine the audio categorization into finer categories. For example, the speech segments may be further categorized as clean speech or speech with Environmental Sounds (EnvS). Similarly, the music segments may be categorized as clean music or music with EnvS. The speech+music segments may also be further refined as only speech+music or speech+music with EnvS. This finer categorization may reveal more intricate relationships between audio types and movie genres. Such additional information may help in further improving the performance of movie genre prediction.

- **Detection of musical instruments for movie genre prediction:** Various musical instruments are used for different types of music used in movies. For example, sentimental scenes in Indian movies might have a soft sitar track playing in the background. On the other hand, a fast-paced scene like car chases or action scenes might be accompanied by a high-tempo drum-set playing in the background. It is not necessary that only a single instrument will be used in all background scores. The distribution of possible musical instruments playing in a particular music segment of a movie might carry information about its genres. Therefore, we plan to explore this direction of feature extraction for movie genre prediction in the future.

- **Classification of environmental sounds for movie genre prediction:** This thesis only

focuses on the speech and music components of movie audio. Such a choice is made as speech and music are the most significant components of movie audio. However, there are other crucial audio components present in movie audio. Various environment sounds and sound effects are frequently found in movies. These types of sounds are used to create the soundscape of a particular scene. For example, a scene showing actors walking might include the sounds of their footsteps. Another scene in a park might have bird-chirping sounds in the background. These kinds of sounds are not directly related to the movie's plot. However, they help in improving the credibility of a visual scene. In other words, environmental sounds may represent the location or time associated with a particular scene. Such information might also have some association with the movie genre. In the future, such aspects of the movie audio can also be explored for their genre prediction.

- **Development of a cultural domain invariant genre prediction system:** The generalization performance reported in chapter 6 indicates that the performance of the movie genre prediction system trained on one domain (Hollywood movies) drops when tested with movie trailers from another domain (Bollywood movies). This issue might be because of the cultural differences between the different movie industries. However, some objective properties of various genres may be consistent across industries. Further research may be done to identify such domain invariant characteristics to develop robust genre prediction systems.

Bibliography

[1] M. Bhattacharjee, S. R. M. Prasanna, and P. Guha, "Speech/Music Classification Using Features From Spectral Peaks," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 28, pp. 1549–1559, 2020.

[2] D. Doukhan, E. Lechapt, M. Evrard, and J. Carrive, "INAS MIREX 2018 MUSIC AND SPEECH DETECTION SYSTEM," in *Proc. Music Information Retrieval Evaluation eXchange (MIREX)*, 2018.

[3] M. Sweney, "Amazon wins 49% of new TV streaming customers in run-up to christmas," 2021. [Online]. Available: www.theguardian.com

[4] J. Alexander, "The entire world is streaming more than ever-and its straining the internet," 2020. [Online]. Available: www.theverge.com

[5] C. Public, "Cisco Annu. Internet Rep. (2018-2023) White Paper," 2020. [Online]. Available: www.cisco.com

[6] P. Kulshreshtha and T. Guha, "Dynamic character graph via online face clustering for movie analysis," *Multimedia Tools and Appl.*, vol. 79, no. 43, pp. 33 103–33 118, 2020.

[7] K. Kurzhals, M. John, F. Heimerl, P. Kuznecov, and D. Weiskopf, "Visual Movie Analytics," *IEEE Trans. on Multimedia*, vol. 18, no. 11, pp. 2149–2160, 2016.

[8] J. Zhang, Y. Wang, Z. Yuan, and Q. Jin, "Personalized real-time movie recommendation system: Practical prototype and evaluation," *Tsinghua Science and Technology*, vol. 25, no. 2, pp. 180–191, 2020.

[9] S.-M. Choi, S.-K. Ko, and Y.-S. Han, "A movie recommendation algorithm based on genre correlations," *Expert Syst. with Appl.*, vol. 39, no. 9, pp. 8079–8085, 2012.

[10] K. Lee, J. Park, I. Kim, and Y. Choi, "Predicting movie success with machine learning techniques: ways to improve accuracy," *Information Syst. Frontiers*, vol. 20, no. 3, pp. 577–588, 2018.

[11] Y. Ru, B. Li, J. Liu, and J. Chai, "An effective daily box office prediction model based on deep neural networks," *Cognitive Syst. Res.*, vol. 52, pp. 182–191, 2018.

[12] A. R. Yamghani and F. Zargari, "Compressed domain video abstraction based on i-frame of hevc coded videos," *Circuits, Syst., and Signal Process.*, vol. 38, no. 4, pp. 1695–1716, 2019.

[13] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. Kuo, "Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques," *IEEE Signal Process. Magazine*, vol. 23, no. 2, pp. 79–89, 2006.

[14] Y. Yi and H. Wang, "Multi-modal learning for affective content analysis in movies," *Multimedia Tools and Appl.*, vol. 78, no. 10, pp. 13 331–13 350, 2019.

[15] G. Irie, T. Satou, A. Kojima, T. Yamasaki, and K. Aizawa, "Affective audio-visual words and latent topic driving model for realizing movie affective scene classification," *IEEE Trans. on Multimedia*, vol. 12, no. 6, pp. 523–535, 2010.

[16] A. Brown, J. Huh, A. Nagrani, J. S. Chung, and A. Zisserman, "Playing a Part: Speaker Verification at the movies," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2021, pp. 6174–6178.

[17] Y. Li, S. S. Narayanan, and C.-C. Kuo, "Adaptive speaker identification with audiovisual cues for movie content analysis," *Pattern Recognition Letters*, vol. 25, no. 7, pp. 777–791, 2004.

[18] Y. Li, Y. Zhang, X. Li, M. Liu, W. Wang, and J. Yang, "Acoustic event diarization in TV/movie audios using deep embedding and integer linear programming," *Multimedia Tools and Appl.*, vol. 78, no. 23, pp. 33 999–34 025, 2019.

[19] J. A. Wi, S. Jang, and Y. Kim, "Poster-Based Multiple Movie Genre Classification Using Inter-Channel Features," *IEEE Access*, vol. 8, pp. 66 615–66 624, 2020.

[20] J. Wehrmann and R. C. Barros, "Movie genre classification: A multi-label approach based on convolutions through time," *Applied Soft Computing*, vol. 61, pp. 973–982, 2017.

[21] G. Lu, "Indexing and Retrieval of Audio: A Survey," *Multimedia Tools and Appl.*, vol. 15, no. 3, pp. 269–290, 2001.

[22] K. Bougiatiotis and T. Giannakopoulos, "Enhanced movie content similarity based on textual, auditory and visual information," *Expert Syst. with Appl.*, vol. 96, pp. 86–102, 2018.

[23] J. Wang, B. Li, W. Hu, and O. Wu, "Horror video scene recognition via multiple-instance learning," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2011, pp. 1325–1328.

[24] A. Austin, E. Moore, U. Gupta, and P. Chordia, "Characterization of movie genre based on music score," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2010, pp. 421–424.

[25] R. Hebbar, K. Somandepalli, and S. Narayanan, "Robust speech activity detection in movie audio: Data resources and experimental evaluation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2019, pp. 4105–4109.

[26] S. Zhang, Q. Huang, S. Jiang, W. Gao, and Q. Tian, "Affective visualization and retrieval for music video," *IEEE Trans. on Multimedia*, vol. 12, no. 6, pp. 510–522, 2010.

[27] L. Canini, S. Benini, and R. Leonardi, "Affective recommendation of movies based on selected connotative features," *IEEE Trans. on Circuits and Syst. for Video Technology*, vol. 23, no. 4, pp. 636–647, 2012.

[28] P. Cascante-Bonilla, K. Sitaraman, M. Luo, and V. Ordonez, "Moviescope: Large-scale analysis of movies using multiple modalities," *arXiv preprint arXiv:1908.03180*, 2019.

[29] S. Chaudhuri, J. Roth, D. P. Ellis, A. Gallagher, L. Kaver, R. Marvin, C. Pantofaru, N. Reale, L. G. Reid, K. Wilson *et al.*, "AVA-speech: A densely labeled dataset of speech activity in movies," *arXiv preprint arXiv:1808.00606*, 2018.

[30] M. Sharma, S. Joshi, T. Chatterjee, and R. Hamid, "A Comprehensive Empirical Review Of Modern Voice Activity Detection Approaches For Movies And TV Shows," *Neurocomputing*, vol. 494, pp. 116–131, 2022.

[31] A. Goyal, N. Kumar, T. Guha, and S. S. Narayanan, "A multimodal mixture-of-experts model for dynamic emotion prediction in movies," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2016, pp. 2822–2826.

[32] Y. Yu, S. Luo, S. Liu, H. Qiao, Y. Liu, and L. Feng, "Deep attention based music genre classification," *Neurocomputing*, vol. 372, pp. 84–91, 2020.

[33] F. Li and M. Akagi, "Blind monaural singing voice separation using rank-1 constraint robust principal component analysis and vocal activity detection," *Neurocomputing*, vol. 350, pp. 44–52, 2019.

[34] L. Chen, S. J. Rizvi, and M. T. Ozsu, "Incorporating audio cues into dialog and action scene extraction," in *Proc. Storage and Retrieval for Media Databases*, vol. 5021, 2003, pp. 252–263.

[35] H. D, Y. Ke, and R. Sukthankar, "SOLAR: sound object localization and retrieval in complex audio environments," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 5, 2005, pp. 429–432.

[36] A. Yadav and D. K. Vishwakarma, "A unified framework of deep networks for genre classification using movie trailer," *Applied Soft Computing*, vol. 96, pp. 106 624/1–14, 2020.

[37] S. Wang and Q. Ji, "Video affective content analysis: a survey of state-of-the-art methods," *IEEE Trans. on Affective Computing*, vol. 6, no. 4, pp. 410–430, 2015.

[38] H. L. Wang and L.-F. Cheong, "Affective understanding in film," *IEEE Trans. on Circuits and Syst. for Video Technology*, vol. 16, no. 6, pp. 689–704, 2006.

[39] M. Soleymani, G. Chanel, J. J. Kierkels, and T. Pun, "Affective ranking of movie scenes using physiological signals and content analysis," in *Proc. 2nd ACM Workshop on Multimedia Semantics*, 2008, pp. 32–39.

[40] G. Irie, K. Hidaka, T. Satou, A. Kojima, T. Yamasaki, and K. Aizawa, "Latent topic driving model for movie affective scene classification," in *Proc. 17th ACM Int. Conf. on Multimedia*, 2009, pp. 565–568.

[41] S. Benini, L. Canini, and R. Leonardi, "A connotative space for supporting movie affective recommendation," *IEEE Trans. on Multimedia*, vol. 13, no. 6, pp. 1356–1370, 2011.

[42] M. Xu, C. Xu, X. He, J. S. Jin, S. Luo, and Y. Rui, "Hierarchical affective content analysis in arousal and valence dimensions," *Signal Process.*, vol. 93, no. 8, pp. 2140–2150, 2013.

[43] J. Tarvainen, J. Laaksonen, and T. Takala, "Film mood and its quantitative determinants in different types of scenes," *IEEE Trans. on Affective Computing*, vol. 11, no. 2, pp. 313–326, 2018.

[44] A. J. Reiss Jr and J. A. Roth, *Understanding and Preventing Violence, Vol. 4: Consequences and Control*. National Academy Press, Washington, DC, USA, 1994.

[45] T. Giannakopoulos, A. Pikrakis, and S. Theodoridis, "A multi-class audio classification method with respect to violent content in movies using bayesian networks," in *Proc. IEEE 9th Workshop on Multimedia Signal Process.*, 2007, pp. 90–93.

[46] T. Giannakopoulos, A. Makris, D. Kosmopoulos, S. Perantonis, and S. Theodoridis, "Audio-visual fusion for detecting violent scenes in videos," in *Proc. Hellenic Conf. on Artificial Intelligence*, 2010, pp. 91–100.

[47] F. D. M. d. Souza, G. C. Chvez, E. A. d. Valle Jr., and A. d. A. Araujo, "Violence Detection in Video Using Spatio-Temporal Features," in *Proc. 23rd SIBGRAPI Conf. on Graphics, Patterns and Images*, 2010, pp. 224–230.

[48] L.-H. Chen, H.-W. Hsu, L.-Y. Wang, and C.-W. Su, "Violence detection in movies," in *Proc. 8th Int. Conf. Computer Graphics, Imaging and Visualization*, 2011, pp. 119–124.

[49] E. Acar, F. Hopfgartner, and S. Albayrak, "Violence detection in hollywood movies by the fusion of visual and mid-level audio cues," in *Proc. 21st ACM Int. Conf. on Multimedia*, 2013, pp. 717–720.

[50] M. Sjöberg, Y. Baveye, H. Wang, V. L. Quang, B. Ionescu, E. Dellandréa, M. Schedl, C.-H. Demarty, and L. Chen, "The mediaeval 2015 affective impact of movies task." in *Proc. MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2015.

[51] M. Sjöberg, B. Ionescu, Y.-G. Jiang, V. L. Quang, M. Schedl, C.-H. Demarty *et al.*, "The MediaEval 2014 Affect Task: Violent Scenes Detection," in *Proc. MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2014.

[52] C.-H. Demarty, C. Penet, M. Schedl, I. Bogdan, V. L. Quang, and Y.-G. Jiang, "The MediaEval 2013 Affect Task: Violent Scenes Detection," in *Proc. MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2013.

[53] C.-H. Demarty, C. Penet, G. Gravier, and M. Soleymani, "The MediaEval 2012 Affect Task: Violent Scenes Detection," in *Proc. MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2012.

[54] ——, "The MediaEval 2011 Affect Task: Violent Scenes Detection," in *Proc. MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2011.

[55] M. G. Constantin, L. D. Stefan, B. Ionescu, C.-H. Demarty, M. Sjöberg, M. Schedl, and G. Gravier, "Affect in Multimedia: Benchmarking Violent Scenes Detection," *IEEE Trans. on Affective Computing*, pp. 1–1, 2020.

[56] Q. Dai, Z. Wu, Y.-G. Jiang, X. Xue, and J. Tang, "Fudan-NJUST at MediaEval 2014: Violent Scenes Detection Using Deep Neural Networks." in *Proc. MediaEval Workshop*, 2014.

[57] N. Derbas, B. Safadi, G. Quénot *et al.*, "LIG at MediaEval 2013 Affect Task: Use of a Generic Method and Joint Audio-Visual Words." in *Proc. MediaEval 2011 Workshop*, 2013.

[58] C. Penet, C.-H. Demarty, G. Gravier, and P. Gros, "Technicolor/inria team at the mediaeval 2013 violent scenes detection task," in *Proc. MediaEval Workshop*, 2013.

[59] J. Schlüter, B. Ionescu, I. Mironica, and M. Schedl, "ARF@ MediaEval 2012: An Uninformed Approach to Violence Detection in Hollywood Movies." in *Proc. MediaEval Workshop*, 2012.

[60] Q. Dai, R.-W. Zhao, Z. Wu, X. Wang, Z. Gu, W. Wu, and Y.-G. Jiang, "Fudan-Huawei at MediaEval 2015: Detecting Violent Scenes and Affective Impact in Movies with Deep Learning." in *Proc. MediaEval Workshop*, 2015.

[61] C. C. Tan and C.-W. Ngo, "The Vireo Team at MediaEval 2013: Violent Scenes Detection by Mid-level Concepts Learnt from Youtube," in *Proc. MediaEval Workshop*, 2013.

[62] G. Gninkoun and M. Soleymani, "Automatic violence scenes detection: A multi-modal approach," in *Proc. MediaEval Workshop*, 2011.

[63] B. Safadi and G. Quénot, "LIG at MediaEval 2011 affect task: use of a generic method," in *Proc. MediaEval Workshop*, 2011.

[64] H. Glotin, J. Razik, S. Paris, and J.-M. Prevot, "Real-time entropic unsupervised violent scenes detection in Hollywood movies-DYNI@ MediaEval Affect Task 2011." in *Proc. MediaEval Workshop*, 2011.

[65] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews," in *Proc. Object Recognition Supported by User Interaction for Service Robots*, vol. 2, 2002, pp. 1086–1089.

[66] Z. Rasheed, Y. Sheikh, and M. Shah, "On the use of computable features for film classification," *IEEE Trans. on Circuits and Syst. for Video Technology*, vol. 15, no. 1, pp. 52–64, 2005.

[67] S. K. Jain and R. Jadon, "Movies genres classifier using neural network," in *Proc. 24th Int. Symp. on Computer and Information Sciences*, 2009, pp. 575–580.

[68] H. Zhou, T. Hermans, A. V. Karandikar, and J. M. Rehg, "Movie genre classification via scene categorization," in *Proc. 18th ACM Int. Conf. on Multimedia*, 2010, pp. 747–750.

[69] J. Wang, B. Li, W. Hu, and O. Wu, "Horror movie scene recognition based on emotional perception," in *Proc. IEEE Int. Conf. on Image Process.*, 2010, pp. 1489–1492.

[70] Y.-F. Huang and S.-H. Wang, "Movie genre classification using SVM with audio and video features," in *Proc. Int. Conf. on Active Media Technology*, 2012, pp. 1–10.

[71] C.-M. Wang and Y.-F. Huang, "Self-adaptive harmony search algorithm for optimization," *Expert Systems with Applications*, vol. 37, no. 4, pp. 2826–2837, 2010.

[72] G. S. Simões, J. Wehrmann, R. C. Barros, and D. D. Ruiz, "Movie genre classification with convolutional neural networks," in *Proc. Int. Joint Conf. on Neural Networks (IJCNN)*, 2016, pp. 259–266.

[73] A. Tadimari, N. Kumar, T. Guha, and S. S. Narayanan, "Opening big in box office? trailer content can help," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2016, pp. 2777–2781.

[74] J. Wehrmann, R. C. Barros, G. S. Simões, T. S. Paula, and D. D. Ruiz, "(Deep) learning from frames," in *Proc. 5th Brazilian Conf. on Intelligent Syst. (BRACIS)*, 2016, pp. 1–6.

[75] J. Wehrmann and R. C. Barros, "Convolutions through time for multi-label movie genre classification," in *Proc. Symp. on Applied Computing*, 2017, pp. 114–119.

[76] O. Ben-Ahmed and B. Huet, "Deep multimodal features for movie genre and interestingness prediction," in *Proc. Int. Conf. on Content-based Multimedia Indexing (CBMI)*, 2018, pp. 1–6.

[77] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016.

[78] Y. Aytar, C. Vondrick, and A. Torralba, "Soundnet: Learning sound representations from unlabeled video," vol. 29, 2016, pp. 892–900.

[79] F. Álvarez, F. Sánchez, G. Hernández-Peñaloza, D. Jiménez, J. M. Menéndez, and G. Cisneros, "On the influence of low-level visual features in film classification," *PLOS ONE*, vol. 14, no. 2, pp. 1–29, 02 2019.

[80] W.-T. Chu and H.-J. Guo, "Movie genre classification based on poster images with deep neural networks," in *Proc. Workshop on Multimodal Understanding of Social, Affective and Subjective Attributes (MUSA2'17)*, 2017, pp. 39–45.

[81] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. Advances in Neural Information Process. Syst.*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25, 2012.

[82] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proc. IEEE Conf. on Computer Vision and Pattern Recog.*, 2017, pp. 7263–7271.

[83] P. G. Shambharkar, P. Thakur, S. Imadoddin, S. Chauhan, and M. Doja, "Genre Classification of Movie Trailers using 3D Convolutional Neural Networks," in *Proc. 4th Int. Conf. on Intelligent Computing and Control Syst. (ICICCS)*, 2020, pp. 850–858.

[84] R. B. Mangolin, R. M. Pereira, A. S. Britto, C. N. Silla, V. D. Feltrim, D. Bertolini, and Y. M. Costa, "A multimodal approach for multi-label movie genre classification," *Multimedia Tools and Appl.*, pp. 1–26, 2020.

[85] E. Fish, J. Weinbren, and A. Gilbert, "Rethinking Genre Classification With Fine Grained Semantic Clustering," in *Proc. IEEE Int. Conf. on Image Process. (ICIP)*, 2021, pp. 1274–1278.

[86] R. Altman, "A Semantic/Syntactic Approach to Film Genre," *Cinema J.*, vol. 23, pp. 6–18, 1984.

[87] S. Neale, "Questions of genre," *Film Genre Reader IV*, pp. 178–202, Jul 2012.

[88] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman, "Use what you have: Video retrieval using representations from collaborative experts," *arXiv preprint arXiv:1907.13487*, 2019.

[89] A. Miech, I. Laptev, and J. Sivic, "Learnable pooling with context gating for video classification," *arXiv preprint*, vol. arXiv:1706.06905, 2017.

[90] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," *arXiv preprint arXiv:1609.08675*, 2016.

[91] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, vol. 40, no. 06, pp. 1437–1451, Jun 2018.

[92] A. Sharma, M. Jindal, A. Mittal, and D. K. Vishwakarma, "A Unified Audio Analysis Framework For Movie Genre Classification Using Movie Trailers," in *Proc. Int. Conf. on Emerging Smart Computing and Informatics (ESCI)*, 2021, pp. 510–515.

[93] D. K. Vishwakarma, M. Jindal, A. Mittal, and A. Sharma, "Multilevel profiling of situation and dialogue-based deep networks for movie genre classification using movie trailers," *arXiv*, 2021.

[94] D. Brezeale and D. J. Cook, "Automatic Video Classification: A Survey of the Literature," *IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 3, pp. 416–430, May 2008.

[95] M. Rouvier, S. Oger, G. Linarès, D. Matrouf, B. Merialdo, and Y. Li, "Audio-based Video Genre Identification," *IEEE/ACM Trans. on Audio, Speech and Lang. Process.*, vol. 23, no. 6, pp. 1031–1041, Jun 2015.

[96] S. Moncrieff, C. Dorai, and S. Venkatesh, "Affect computing in film through sound energy dynamics," in *Proc. 9th ACM Int. Conf. on Multimedia*, 2001, pp. 525–527.

[97] D. Bordwell and K. Thompson, *Film Art: An Introduction.*, 8th ed. McGraw-Hill, New York, NY, USA, 2008.

[98] J. Saunders, "Real-time discrimination of broadcast speech/music," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 2, May 1996, pp. 993–996.

[99] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 2, Apr 1997, pp. 1331–1334.

[100] G. Williams and D. P. Ellis, "Speech/music discrimination based on posterior probability features," in *Proc. Eurospeech*, 1999.

[101] T. Zhang and C. C. J. Kuo, "Hierarchical classification of audio data for archiving and retrieving," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process.. Proc. (ICASSP)*, vol. 6, Mar 1999, pp. 3001–3004.

[102] K. El-Maleh, M. Klein, G. Petrucci, and P. Kabal, "Speech/music discrimination for multimedia appl." in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 6, 2000, pp. 2445–2448.

[103] P. J. Moreno and R. Rifkin, "Using the Fisher kernel method for Web audio classification," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 6, 2000, pp. 2417–2420 vol.4.

[104] T. Zhang and C.-C. J. Kuo, "Audio content analysis for online audiovisual data segmentation and classification," *IEEE Trans. on speech and audio processing*, vol. 9, no. 4, pp. 441–457, 2001.

[105] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Trans. on Speech and Audio Process.*, vol. 10, no. 7, pp. 504–516, Oct 2002.

[106] A. Bugatti, A. Flammini, and P. Migliorati, "Audio Classification in Speech and Music: A Comparison between a Statistical and a Neural Approach," *EURASIP J. on Advances in Signal Process.*, no. 4, p. 980905, Apr 2002.

[107] J. Pinquier, J.-L. Rouas, and R. André-Obrecht, "Robust speech / music classification in audio documents," in *Proc. 7th Int. Conf. on Spoken Lang. Process. (ICSLP)*, Denver, Colorado, USA, Sep 2002.

[108] J. Pinquier, J. L. Rouas, and R. André-Obrecht, "A fusion study in speech/music classification," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 2, Apr 2003, pp. 17–20.

[109] J. Ajmera, I. McCowan, and H. Bourlard, "Speech/music segmentation using entropy and dynamism features in a hmm classification framework," *Speech Commun.*, vol. 40, no. 3, pp. 351 – 363, 2003.

[110] J. J. Burred and A. Lerch, "Hierarchical automatic audio signal classification," *J. Audio Eng. Soc*, vol. 52, no. 7/8, pp. 724–739, 2004.

[111] E. Alexandre-Cortizo, M. Rosa-Zurera, and F. Lopez-Ferreras, "Application of fisher linear discriminant analysis to speech/music classification," in *Proc. EUROCON 2005-The Int. Conf. on "Computer as a Tool"*, vol. 2, Nov 2005, pp. 1666–1669.

[112] C. Panagiotakis and G. Tziritas, "A speech/music discriminator based on RMS and zero-crossings," *IEEE Trans. on Multimedia*, vol. 7, no. 1, pp. 155–166, Feb 2005.

[113] J. Keum and H. Lee, "Speech/Music Discrimination using Spectral Peak Feature for Speaker Indexing," in *Proc. Int. Symp. on Intelligent Signal Process. and Commun.*, Dec 2006, pp. 323–326.

[114] ——, "Speech/Music Discrimination Based on Spectral Peak Analysis and Multi-layer Perceptron," in *Proc. Int. Conf. on Hybrid Information Technology*, vol. 2, Nov 2006, pp. 56–61.

[115] N. Mesgarani, M. Slaney, and S. A. Shamma, "Discrimination of speech from nonspeech based on multiscale spectro-temporal Modulations," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 14, no. 3, pp. 920–930, May 2006.

[116] J. E. Muñoz-Exposito, S. Garcia-Galán, N. Ruiz-Reyes, P. Vera-Candeas, and F. Rivas-Peña, "Speech/music discrimination using awarped lpc-based feature and a fuzzy expert system for intelligent audio coding," in *Proc. 14th European Signal Process. Conf.*, Sept 2006, pp. 1–5.

[117] S. Garcia Gálan, J. E. Muñoz-Exposito, F. Rivas Peña, N. Ruiz-Reyes, and P. Vera-Candeas, "A Fuzzy Rules-based Speech/Music Discrimination Approach for Intelligent Audio Coding Over the Internet," in *Proc. Audio Engineering Soc. Conv. 120*, May 2006.

[118] J. G. A. Barbedo and A. Lopes, "A robust and computationally efficient speech/music discriminator," *J. Audio Eng. Soc.*, vol. 54, no. 7/8, pp. 571–588, 2006.

[119] G. Farahania, S. M. Ahadia, and M. M. Homayounpourb, "Features based on filtering and spectral peaks in autocorrelation domain for robust speech recognition," *Computer Speech & Lang.*, vol. 21, no. 1, pp. 187 – 205, 2007.

[120] J. H. Song, K. H. Lee, J. H. Chang, J. K. Kim, and N. S. Kim, "Analysis and Improvement of Speech/Music Classification for 3GPP2 SMV Based on GMM," *IEEE Signal Process. Letters*, vol. 15, pp. 103–106, 2008.

[121] A. Pikrakis, T. Giannakopoulos, and S. Theodoridis, "A Speech/Music Discriminator of Radio Recordings Based on Dynamic Programming and Bayesian Networks," *IEEE Trans. on Multimedia*, vol. 10, no. 5, pp. 846–857, Aug 2008.

[122] K. Seyerlehner, T. Pohle, M. Schedl, and G. Widmer, "Automatic music detection in television productions," in *Proc. 10th Int. Conf. on Digital Audio Effects (DAFx-10)*, vol. 10, Bordeaux, France, Sep 2007.

[123] K. Lee and D. P. Ellis, "Detecting music in ambient audio by long-window autocorrelation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2008, pp. 9–12.

[124] T. Izumitani, R. Mukai, and K. Kashino, "A background music detection method based on robust feature extraction," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2008, pp. 13–16.

[125] J. Anemüller, D. Schmidt, and J.-H. Bach, "Detection of speech embedded in real acoustic background based on amplitude modulation spectrogram features," in *Proc. Interspeech*, 2008.

[126] T. Taniguchi, M. Tohyama, and K. Shirai, "Detection of speech and music based on spectral tracking," *Speech Commun.*, vol. 50, no. 7, pp. 547–563, 2008.

[127] Y. Lavner and D. Ruinskiy, "A decision-tree-based algorithm for speech/music classification and segmentation," *EURASIP J. on Audio, Speech, and Music Process.*, vol. 2009, no. 1, p. 239892, Jun 2009.

[128] A. Gallardo-Antolin and J. M. Montero, "Histogram Equalization-Based Features for Speech, Music, and Song Discrimination," *IEEE Signal Process. Letters*, vol. 17, no. 7, pp. 659–662, Jul 2010.

[129] J. C. Segura, C. Benitez, A. de la Torre, A. J. Rubio, and J. Ramirez, "Cepstral domain segmental nonlinear feature transformations for robust speech recognition," *IEEE Signal Process. Letters*, vol. 11, no. 5, pp. 517–520, May 2004.

[130] S.-H. Lin, Y.-M. Yeh, and B. Chen, "Exploiting polynomial-fit histogram equalization and temporal average for robust speech recognition," in *Proc. Int. Conf. on Spoken Lang. Process.*, 2006, p. 25222525.

[131] L. J. Tardón, S. Sammartino, and I. Barbancho, "Design of an efficient music-speech discriminator," *The J. of the Acoustical Soc. of America*, vol. 127, no. 1, pp. 271–279, 2010.

[132] J.-H. Bach, J. Anemüller, and B. Kollmeier, "Robust speech detection in real acoustic backgrounds with perceptually motivated features," *Speech Commun.*, vol. 53, no. 5, pp. 690–706, 2011.

[133] F. Weninger, J. Feliu, and B. Schuller, "Supervised and semi-supervised suppression of background music in monaural speech recordings," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2012, pp. 61–64.

[134] C. Lim and J. h. Chang, "Enhancing support vector machine-based speech/music classification using conditional maximum a posteriori criterion," *IET Signal Process.*, vol. 6, no. 4, pp. 335–340, Jun 2012.

[135] J. C. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," in *Proc. Advances in Large Margin Classifiers*, 1999, pp. 61–74.

[136] C. Lim, S. R. Lee, and J. H. Chang, "Efficient implementation of an SVM-based speech/music classifier by enhancing temporal locality in support vector references," *IEEE Trans. on Consumer Electronics*, vol. 58, no. 3, pp. 898–904, Aug 2012.

[137] M. Srinivas, D. Roy, and C. K. Mohan, "Learning Sparse Dictionaries for Music and Speech Classification," in *Proc. 19th Int. Conf. on Digital Signal Process.*, Aug 2014, pp. 673–675.

[138] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. 26th Annu. Int. Conf. on Machine Learning*, ser. ICML '09, New York, NY, USA, 2009, pp. 689–696.

[139] P. Neammalai, S. Phimoltares, and C. Lursinsap, "Speech and music classification using hybrid form of spectrogram and fourier transformation," in *Proc. Signal and Information Process. Assoc. Annu. Summit and Conf. (APSIPA)*, Dec 2014, pp. 1–6.

[140] G. Sell and P. Clark, "Music tonality features for speech/music discrimination," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, May 2014, pp. 2489–2493.

[141] D. Castán, A. Ortega, A. Miguel, and E. Lleida, "Audio segmentation-by-classification approach based on factor analysis in broadcast news domain," *EURASIP J. on Audio, Speech, and Music Process.*, vol. 2014, no. 1, p. 34, 2014.

[142] B. K. Khonglah and S. R. M. Prasanna, "Speech / music classification using Vocal Tract Constriction aspect of speech," in *Proc. Annu. IEEE India Conf. (INDICON)*, Dec 2015, pp. 1–6.

[143] B. D. Sarma and S. R. M. Prasanna, "Analysis of vocal tract constrictions using zero frequency filtering," *IEEE Signal Process. Letters*, vol. 21, no. 12, pp. 1481–1485, Dec 2014.

[144] H. Zhang, X.-K. Yang, W.-Q. Zhang, W.-L. Zhang, and J. Liu, "Application of i-vector in speech and music classification," in *Proc. IEEE Int. Symp. on Signal Process. and Information Technology (ISSPIT)*, Dec 2016, pp. 1–5.

[145] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech and Audio Process.*, vol. 3, no. 1, pp. 72–83, 1995.

[146] D. G. Stork, R. O. Duda, P. E. Hart, and D. Stork, "Pattern Classification," *A Wiley-Interscience Publication*, 2001.

[147] B. K. Khonglah and S. R. M. Prasanna, "Low Frequency Region of Vocal Tract Information for Speech / Music Classification," in *Proc. IEEE Region 10 Conf. (TENCON)*, Nov 2016, pp. 2593–2597.

[148] B. K. Khonglah and S. R. Mahadeva Prasanna, "Speech/Music Classification Using Speech-specific Features," *Digital Signal Process.*, vol. 48, no. C, pp. 71–83, Jan 2016.

[149] E. Mezghani, M. Charfeddine, C. B. Amar, and H. Nicolas, "Multifeature speech/music discrimination based on mid-term level statistics and supervised classifiers," in *Proc. IEEE/ACS 13th Int. Conf. of Computer Syst. and Appl. (AICCSA)*, Nov 2016, pp. 1–8.

[150] P. Lopez-Otero, L. Docio-Fernandez, and C. Garcia-Mateo, "Ensemble audio segmentation for radio and television programmes," *Multimedia Tools and Appl.*, vol. 76, no. 5, pp. 7421–7444, Mar 2017.

[151] J. Vavrek, P. Fecilák, J. Juhár, and A. Čižmár, "Classification of broadcast news audio data employing binary decision architecture," *Computing and Informatics*, vol. 36, no. 4, pp. 857–886, 2017.

[152] J. Vavrek, E. Vozáriková, M. Pleva, and J. Juhár, "Broadcast news audio classification using svm binary trees," in *Proc. 35th IEEE Int. Conf. on TeleCommun.s and Signal Process.*, 2012, pp. 469–473.

[153] D. Y. Mohammed and F. F. Li, "Overlapped soundtracks segmentation using singular spectrum analysis and random forests," in *Proc. 2nd Int. Conf. on Knowledge Engineering and Appl. (ICKEA)*, 2017, pp. 49–54.

[154] P. Gimeno, V. Mingote, A. Ortega, A. Miguel, and E. Lleida, "Partial auc optimisation using recurrent neural networks for music detection with limited training data," in *Proc. Interspeech*, 2020, pp. 3067–3071.

[155] B. Jia, J. Lv, X. Peng, Y. Chen, and S. Yang, "Hierarchical Regulated Iterative Network for Joint Task of Music Detection and Music Relative Loudness Estimation," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 29, pp. 1–13, 2021.

[156] R. André-Obrecht, "A new statistical approach for the automatic segmentation of continuous speech signals," *IEEE Trans. on Acoust., Speech, and Signal Process.*, vol. 36, no. 1, pp. 29–40, Jan 1988.

[157] Z. Liu, J. Huang, Y. Wang, and T. Chen, "Audio feature extraction and analysis for scene classification," in *Proc. 1st Signal Process. Soc. Workshop on Multimedia Signal Process.*, Jun 1997, pp. 343–348.

[158] J. Foote, "A similarity measure for automatic audio classification," 1997.

[159] W. M. Fisher, "The DARPA speech recognition research database : Specifications and status," *Proc. DARPA Workshop Speech Recognition*, pp. 93–99, 1986.

[160] J. Ajmera, I. A. McCowan, and H. Bourlard, "Robust HMM-based speech/music segmentation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 1, May 2002, pp. I–297–I–300.

[161] R. McAulay and T. Quatieri, "Speech analysis/Synthesis based on a sinusoidal representation," *IEEE Trans. on Acoust., Speech, and Signal Process.*, vol. 34, no. 4, pp. 744–754, Aug 1986.

[162] J. O. Smith and X. Serra, "PARSHL: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation," in *Proc. Int. Computer Music Conf. (ICMC)*, Aug 23-26 1987, pp. 290–7.

[163] M. Padmanabhan, "Spectral peak tracking and its use in speech recognition," in *Proc. 6th Int. Conf. on Spoken Lang. Process. (ICSLP)*, vol. 4, Beijing, China, Oct 2000, pp. 604–607.

[164] M. Lagrange, S. Marand, and J. Rault, "Enhancing the Tracking of Partials for the Sinusoidal Modeling of Polyphonic Sounds," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 15, no. 5, pp. 1625–1634, Jul 2007.

[165] M. J. Prerau, P. L. Purdon, and U. T. Eden, "Tracking non-stationary spectral peak structure in eeg data," in *Proc. 35th Annu. Int. Conf. of the IEEE Engineering in Medicine and Biology Soc. (EMBC)*, Jul 2013, pp. 417–420.

[166] Z. Zhang, Z. Pi, and B. Liu, "TROIKA: A General Framework for Heart Rate Monitoring Using Wrist-Type Photoplethysmographic Signals During Intensive Physical Exercise," *IEEE Trans. on Biomed. Engineering*, vol. 62, no. 2, pp. 522–531, Feb 2015.

[167] N. K. L. Murthy, P. C. Madhusudana, P. Suresha, V. Periyasamy, and P. K. Ghosh, "Multiple Spectral Peak Tracking for Heart Rate Monitoring from Photoplethysmography Signal During Intensive Physical Exercise," *IEEE Signal Process. Letters*, vol. 22, no. 12, pp. 2391–2395, Dec 2015.

[168] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. on Acoust., Speech, and Signal Process.*, vol. 27, no. 2, pp. 113–120, Apr 1979.

[169] O. Cappe, "Elimination of the musical noise phenomenon with the ephraim and malah noise suppressor," *IEEE Trans. on Speech and Audio Process.*, vol. 2, no. 2, pp. 345–349, Apr 1994.

[170] P. Gopalakrishnan, D. Nahamoo, M. Panmanabhan, and L. Polymenakos, "Method and apparatus for suppressing background music or noise from the speech input of a speech recognizer," Dec 1998, uS Patent 5,848,163.

[171] Zenton Goh, Kah-Chye Tan, and T. G. Tan, "Postprocessing method for suppressing musical noise generated by spectral subtraction," *IEEE Trans. on Speech and Audio Process.*, vol. 6, no. 3, pp. 287–292, May 1998.

[172] N. Virag, "Single channel speech enhancement based on masking properties of the human auditory system," *IEEE Trans. on Speech and Audio Process.*, vol. 7, no. 2, pp. 126–137, Mar 1999.

[173] T. Esch and P. Vary, "Efficient musical noise suppression for speech enhancement system," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Apr 2009, pp. 4409–4412.

[174] C. Breithaupt, T. Gerkmann, and R. Martin, "Cepstral smoothing of spectral filter gains for speech enhancement without musical noise," *IEEE Signal Process. Letters*, vol. 14, no. 12, pp. 1036–1039, Dec 2007.

[175] T. Hasan and M. K. Hasan, "Suppression of Residual Noise From Speech Signals Using Empirical Mode Decomposition," *IEEE Signal Process. Letters*, vol. 16, no. 1, pp. 2–5, Jan 2009.

[176] K. V. Sørensen and S. V. Andersen, "Speech enhancement with natural sounding residual noise based on connected time-frequency speech presence regions," *EURASIP J. on Advances in Signal Process.*, vol. 2005, no. 18, p. 305909, Nov 2005.

[177] A. A. Nugraha, A. Liutkus, and E. Vincent, "Multichannel Audio Source Separation With Deep Neural Networks," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 24, no. 9, pp. 1652–1664, Sep 2016.

[178] D. Kounades-Bastian, L. Girin, X. Alameda-Pineda, S. Gannot, and R. Horaud, "An EM algorithm for joint source separation and diarisation of multichannel convolutive speech mixtures," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2017, pp. 16–20.

[179] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A Consolidated Perspective on Multimicrophone Speech Enhancement and Source Separation," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 25, no. 4, pp. 692–730, Apr 2017.

[180] E. Cano, D. FitzGerald, A. Liutkus, M. D. Plumbley, and F. Stter, "Musical Source Separation: An Introduction," *IEEE Signal Process. Magazine*, vol. 36, no. 1, pp. 31–40, Jan 2019.

[181] B. Raj, T. Virtanen, S. Chaudhuri, and R. Singh, "Non-negative matrix factorization based compensation of music for automatic speech recognition," in *Proc. INTERSPEECH*, Sep 2010, pp. 717–720.

[182] J. Droppo and A. Acero, "Noise robust speech recognition with a switching linear dynamic model," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 1, May 2004, pp. I–953.

[183] B. Raj, R. Singh, and R. Stern, "On tracking noise with linear dynamical system models," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 1, May 2004, pp. I–965.

[184] A. P. Varga and R. K. Moore, "Hidden markov model decomposition of speech and noise," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Apr 1990, pp. 845–848.

[185] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: A graphical modeling approach," *Computer Speech & Lang.*, vol. 24, no. 1, pp. 45 – 66, 2010.

[186] Y. Luo, Z. Chen, J. R. Hershey, J. L. Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: Stronger together," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2017, pp. 61–65.

[187] P. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, Dec 2015.

[188] J. Sebastian and H. A. Murthy, "Group delay based music source separation using deep recurrent neural networks," in *Proc. Int. Conf. on Signal Process. and Commun. (SPCOM)*, Jun 2016, pp. 1–5.

[189] B. Yegnanarayana, H. A. Murthy, and V. R. Ramachandran, "Process. of noisy speech using modified group delay functions," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 2, Apr 1991, pp. 945–948.

[190] E. M. Grais and M. D. Plumbley, "Single channel audio source separation using convolutional denoising autoencoders," *Proc. IEEE Global Conf. on Signal and Information Process. (GlobalSIP)*, pp. 1265–1269, 2017.

[191] S. Uhlich, M. Porcu, F. Giron, M. Enenkl, T. Kemp, N. Takahashi, and Y. Mitsufuji, "Improving music source separation based on deep neural networks through data augmentation and network blending," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2017, pp. 261–265.

[192] D. Yu, M. Kolbk, Z. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2017, pp. 241–245.

[193] R. Kumar, Y. Luo, and N. Mesgarani, "Music source activity detection and separation using deep attractor network," in *Proc. Interspeech*, 2018, pp. 347–351.

[194] J. Muth, S. Uhlich, N. Perraudin, T. Kemp, F. Cardinaux, and Y. Mitsufuji, "Improving DNN-based Music Source Separation using Phase Features," *arXiv*, vol. abs/1807.02710, 2018.

[195] Y. C. Subakan and P. Smaragdis, "Generative Adversarial Source Separation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Apr 2018, pp. 26–30.

[196] P. Seetharaman, G. Wichern, J. L. Roux, and B. Pardo, "Bootstrapping deep music separation from primitive auditory grouping principles," *arXiv*, vol. 1910.11133, 2019.

[197] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 1066–1074, Mar 2007.

[198] M. Shashanka, B. Raj, and P. Smaragdis, "Probabilistic Latent Variable Models as Nonnegative Factorizations," *Computational Intelligence and Neuroscience*, vol. 2008, May 2008.

[199] S. Abdali and B. NaserSharif, "Non-negative matrix factorization for speech/music separation using source dependent decomposition rank, temporal continuity term and filtering," *Biomed. Signal Process. and Control*, vol. 36, pp. 168–175, 2017.

[200] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2016, pp. 31–35.

[201] E. Fish, J. Weinbren, and A. Gilbert, "Rethinking movie genre classification with fine-grained semantic clustering," *arXiv preprint arXiv:2012.02639*, 2020.

[202] Y. Deldjoo, M. G. Constantin, B. Ionescu, M. Schedl, and P. Cremonesi, "MMTF-14K: a multifaceted movie trailer feature dataset for recommendation and retrieval," in *Proc. 9th ACM Multimedia Syst. Conf.*, 2018, pp. 450–455.

[203] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv*, 2015.

[204] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. on Speech and Audio Process.*, vol. 10, no. 5, pp. 293–302, Jul 2002.

[205] D. Tidhar, D. Wolff, T. Weyde, and E. Benetos, "Muspeak: Automatic segmentation of audio recordings to speech and music," City Univ. London Res. Pump Priming Fund, 2015. [Online]. Available: https://mirg.city.ac.uk/muspeak

[206] J. Schlüter and R. Sonnleitner, "Unsupervised Feature Learning for Speech and Music Detection in Radio Broadcasts," in *Proc. 15th Int. Conf. on Digital Audio Effects (DAFx-12)*, vol. 15, York, UK, 2012.

[207] F. Kurth and M. Muller, "Efficient Index-Based Audio Matching," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 16, no. 2, pp. 382–395, Feb 2008.

[208] V. A. Masoumeh and M. B. Mohammad, "A Review on Speech-Music Discrimination Methods," *Int. J. of Computer Science and Network Solutions*, vol. 2, pp. 67–78, Feb 2014.

[209] A. Kruspe, D. Zapf, and H. Lukashevich, "Automatic speech/music discrimination for broadcast signals," in *Proc. INFORMATIK*, 2017, pp. 151–162.

[210] A. Pikrakis and S. Theodoridis, "Speech-music discrimination: A deep learning perspective," in *Proc. 22nd European Signal Process. Conf. (EUSIPCO)*, Sep 2014, pp. 616–620.

[211] D. Doukhan and J. Carrive, "Investigating the Use of Semi-Supervised Convolutional Neural Network Models for Speech/Music Classification and Segmentation," in *Proc. 9th Int. Conf. on Advances in Multimedia (MMEDIA 2017) :* , ser. MMEDIA 2017 : The 9th Int. Conf. on Advances in Multimedia, Venise, Italy, Apr 2017.

[212] M. Papakostas and T. Giannakopoulos, "Speech-music discrimination using deep visual feature extractors," *Expert Syst. with Appl.*, vol. 114, pp. 334–344, 2018.

[213] S. Cheung and J. S. Lim, "Combined multi-resolution (wideband/narrowband) spectrogram," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 1, Apr 1991, pp. 457–460.

[214] X. Xu, Yiand Sun, "Maximum speed of pitch change and how it may relate to speech," *The J. of the Acoustical Soc. of America*, vol. 111, no. 3, pp. 1399–1413, 2002.

[215] J. F. Alm and J. S. Walker, "Time-frequency analysis of musical instruments," *Soc. for Industrial and Applied Mathematics Review*, vol. 44, no. 3, pp. 457–476, Aug 2002.

[216] M. J. Hawley, "Structure out of sound," Ph.D. dissertation, Massachusetts Inst. of Technology, 1993.

[217] Z. Zhang, "Mechanics of human voice production and control," *The J. of the Acoustical Soc. of America*, vol. 140(4), pp. 2614–2635, 2016.

[218] K. S. R. Murty and B. Yegnanarayana, "Epoch Extraction From Speech Signals," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 16, no. 8, pp. 1602–1613, Nov 2008.

[219] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *IEEE Trans. on Speech and Audio Process.*, vol. 13, no. 5, pp. 1035–1047, Sept 2005.

[220] J. Meyer, *Structure of Musical Sound.* New York: Springer-Verlag, New York; Berlin, Germany; Vienna, Austria, 2009.

[221] L. L. Oller, *Analysis of Voice Signals for the Harmonics-to-noise Crossover Frequency.* UPC, Barcelona, Spain: KTH Royal Inst. of Technology, 2008.

[222] P. Boersma and D. Weenink, "Praat: doing phonetics by computer [Computer program]," Mar 2018. [Online]. Available: http://www.praat.org/

[223] J. Urbano, D. Bogdanov, H. Boyer, E. Gómez Gutiérrez, X. Serra *et al.*, "What is the effect of audio quality on the robustness of MFCCs and chroma features?" in *Proc. 15th Conf. of the Int. Soc. for Music Information Retrieval (ISMIR)*, Taipei, Taiwan, Oct 2014, pp. 573–578.

[224] X. Dong, M. Bocko, and Z. Ignjatovic, "Data hiding via phase manipulation of audio signals," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 5, 2004, pp. V–377.

[225] M. Choudhury, R. Bhagwan, and K. Bali, "The Use Of Melodic Scales In Bollywood Music: An Empirical Study." in *Proc. Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 2013, pp. 59–64.

[226] M. Mukherjee, "The architecture of songs and music: soundmarks of Bollywood, a popular form and its emergent texts," *Screen Sound J.*, vol. 3, pp. 9–34, 2012.

[227] C. Glaser, M. Heckmann, F. Joublin, and C. Goerick, "Combining Auditory Preprocessing and Bayesian Estimation for Robust Formant Tracking," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 18, no. 2, pp. 224–236, Feb 2010.

[228] L. Rabiner and R. Schafer, *Theory and Appl. of Digital Speech Process.*, 1st ed. Upper Saddle River, NJ, USA: Prentice-Hall, Englewood Cliffs, NJ, USA, 2010.

[229] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition.* Upper Saddle River, NJ, USA: Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[230] E. Zhang and Y. Zhang, *F-Measure*, L. LIU and M. T. ÖZSU, Eds. Boston, MA: US, 2009.

[231] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.

[232] W. Krzanowski, *Principles of multivariate analysis.* Oxford Univ. Press, London, U.K., 2000, vol. 23.

[233] L. Sun, B. Zou, S. Fu, J. Chen, and F. Wang, "Speech emotion recognition based on dnn-decision tree svm model," *Speech Commun.*, vol. 115, pp. 29–37, 2019.

[234] R. Hebbar, K. Somandepalli, and S. Narayanan, "Robust Speech Activity Detection in Movie Audio: Data Resources and Experimental Evaluation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2019, pp. 4105–4109.

[235] A. Goyal, N. Kumar, T. Guha, and S. S. Narayanan, "A multimodal mixture-of-experts model for dynamic emotion prediction in movies," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar 2016, pp. 2822–2826.

[236] K. Łopatka, "Detection of dialogue in movie soundtrack for speech intelligibility enhancement," in *Proc. Int. Conf. on Multimedia Commun., Services and Security*, 2014, pp. 149–158.

[237] S. W. Smith *et al.*, *The scientist and engineer's guide to digital signal processing.* California Tech. Pub., San Diego, 1997, vol. 14.

[238] M. Triki and D. Slock, "Multi-channel mono-path periodic signal extraction with global amplitude and phase modulation for music and speech signal analysis," in *Proc. IEEE/SP 13th Workshop on Statistical Signal Process.*, 2005, pp. 77–82.

[239] S. Dalla Bella, A. Białuńska, and J. Sowiński, "Why Movement Is Captured by Music, but Less by Speech: Role of Temporal Regularity," *PLOS ONE*, vol. 8, no. 8, 08 2013.

[240] A. Prodeus, V. Didkovskyi, M. Didkovska, and I. Kotvytskyi, "On peculiarities of evaluating the quality of speech and music signals subjected to phase distortion," in *Proc. IEEE 37th Int. Conf. on Electronics and Nanotechnology (ELNANO)*, 2017, pp. 455–460.

[241] I. V. Kotvytskyi and A. M. Prodeus, "Objective and subjective evaluation of the quality of speech and music signals subjected to phase distortions," *Electronics and Commun.*, vol. 21, no. 2, pp. 25–31, 2016.

[242] S. Mukherjee, S. K. Palit, S. Banerjee, M. Ariffin, and D. Bhattacharya, "Phase synchronization of instrumental music signals," *The European Physical J. Special Topics*, vol. 223, no. 8, pp. 1561–1577, 2014.

[243] Y. Bayya and D. N. Gowda, "Spectro-temporal analysis of speech signals using zero-time windowing and group delay function," *Speech Commun.*, vol. 55, no. 6, pp. 782–795, 2013.

[244] H. A. Murthy and V. Gadde, "The modified group delay function and its application to phoneme recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 1, Apr 2003, pp. I–68.

[245] K. Vijayan, P. R. Reddy, and K. S. R. Murty, "Significance of analytic phase of speech signals in speaker verification," *Speech Commun.*, vol. 81, pp. 54–71, 2016.

[246] B. Yegnanarayana, "Formant extraction from linearprediction phase spectra," *The J. of the Acoustical Soc. of America*, vol. 63, no. 5, pp. 1638–1640, 1978.

[247] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[248] J. M. Anand, S. Guruprasad, and B. Yegnanarayana, "Extracting formants from short segments of speech using group delay functions," in *Proc. 9th Int. Conf. on Spoken Lang. Process.*, 2006.

[249] B. K. Khonglah and S. R. Prasanna, "Clean Speech/Speech with Background Music Classification Using HNGD Spectrum," *Int. J. of Speech Technology*, vol. 20, no. 4, p. 10231036, Dec 2017.

[250] S. Sigurdsson, K. B. Petersen, and T. Lehn-Schiøler, "Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music," in *Proc. Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 2006, pp. 286–289.

[251] L. Marple, "Computing the discrete-time "analytic" signal via fft," *IEEE Trans. on Signal Process.*, vol. 47, no. 9, pp. 2600–2603, Sep 1999.

[252] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proc of The 14th Python in Science Conf.*, vol. 8, 2015, pp. 18–25.

[253] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *J. of Machine Learning Res.*, vol. 12, pp. 2825–2830, 2011.

[254] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Conf. on Operating Syst. Design and Implementation*, ser. OSDI'16, USA, 2016, p. 265283.

[255] B. Thompson, "Canonical correlation analysis," *Encyclopedia of statistics in behavioral science*, 2005.

[256] M. E. Sargin, Y. Yemez, E. Erzin, and A. M. Tekalp, "Audiovisual synchronization and fusion using canonical correlation analysis," *IEEE Trans. on Multimedia*, vol. 9, no. 7, pp. 1396–1403, 2007.

[257] B. Meléndez-Catalán, E. Molina, and E. Gómez, "Open broadcast media audio from TV: A dataset of TV broadcast audio with relative music loudness annotations, journal=Trans. of the Int. Soc. for Music Information Retrieval," vol. 2, no. 1, 2019.

[258] *Music Information Retrieval Evaluation eXchange (MIREX)*, 2018. [Online]. Available: www.music-ir.org/mirex/wiki/2018:Music_and_or_Speech_Detection_Results

[259] M. Hussain, M. A. Haque *et al.*, "Swishnet: A fast convolutional neural network for speech, music and noise classification and segmentation," *arXiv preprint arXiv:1812.00149*, 2018.

[260] Z. Li, X. Xie, J. Wang, V. Grancharov, and W. Liu, "Optimization of EVS Speech/Music Classifier based on Deep Learning," in *Proc. 14th IEEE Int. Conf. on Signal Process. (ICSP)*, 2018, pp. 260–264.

[261] G. K. Birajdar and M. D. Patil, "Speech/music classification using visual and spectral chromagram features," *J. of Ambient Intelligence and Humanized Computing*, vol. 11, no. 1, pp. 329–347, 2020.

[262] Q. Lemaire and A. Holzapfel, "Temporal convolutional networks for speech and music detection in radio broadcast," in *Proc. Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, Nov 2019.

[263] B.-Y. Jang, W.-H. Heo, J.-H. Kim, and O.-W. Kwon, "Music detection from broadcast contents using convolutional neural networks with a mel-scale kernel," *EURASIP J. on Audio, Speech, and Music Process.*, vol. 2019, no. 1, pp. 1–12, 2019.

[264] B. Raj, V. N. Parikh, and R. M. Stern, "The effects of background music on speech recognition accuracy," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, vol. 2, 1997, pp. 851–854.

[265] P. Vanroose, "Blind source separation of speech and background music for improved speech recognition," in *Proc. 24th Symp. on Information Theory*, 2003, pp. 103–108.

[266] N. Tsipas, L. Vrysis, C. Dimoulas, and G. Papanikolaou, "Efficient audio-driven multimedia indexing through similarity-based speech/music discrimination," *Multimedia Tools and Appl.*, vol. 76, no. 24, pp. 25 603–25 621, 2017.

[267] S. Venkatesh, D. Moffat, A. Kirke, G. Shakeri, S. Brewster, J. Fachner, H. Odell-Miller, A. Street, N. Farina, S. Banerjee, and E. R. Miranda, "Artificially synthesising data for audio classification and segmentation to improve speech and music detection in radio broadcast," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2021, pp. 636–640.

[268] M. Bhattacharjee, S. R. M. Prasanna, and P. Guha, "Detection of speech overlapped with low-energy music using pyknograms," in *Proc. Nat. Conf. on Commun. (NCC)*, 2021, pp. 1–6.

[269] A. Ortega, D. Castan, A. Miguel, and E. Lleida, "The albayzin 2012 audio segmentation evaluation," in *Proc. IberSpeech*, Madrid, Spain, 2012, pp. 21–23.

[270] D. Castán, D. Tavarez, P. Lopez-Otero, J. Franco-Pedroso, H. Delgado, E. Navas, L. Docio-Fernández, D. Ramos, J. Serrano, A. Ortega *et al.*, "Albayzín-2014 evaluation: audio segmentation and classification in broadcast news domains," *EURASIP J. on Audio, Speech, and Music Process.*, vol. 2015, no. 1, pp. 1–9, 2015.

[271] P. Gimeno, I. Viñals, A. Ortega, A. Miguel, and E. Lleida, "Multiclass audio segmentation based on recurrent neural networks for broadcast domain data," *EURASIP J. on Audio, Speech, and Music Process.*, vol. 2020, no. 1, pp. 1–19, 2020.

[272] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2017, pp. 131–135.

[273] J. S. Gómez, J. Abeßer, and E. Cano, "Jazz Solo Instrument Classification with Convolutional Neural Networks, Source Separation, and Transfer Learning." in *Proc. Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 2018, pp. 577–584.

[274] J. Driedger, M. Müller, and S. Ewert, "Improving Time-Scale Modification of Music Signals Using Harmonic-Percussive Separation," *IEEE Signal Process. Letters*, vol. 21, no. 1, pp. 105–109, 2014.

[275] E. Tsunoo, G. Tzanetakis, N. Ono, and S. Sagayama, "Beyond Timbral Statistics: Improving Music Classification Using Percussive Patterns and Bass Lines," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 19, no. 4, pp. 1003–1014, 2011.

[276] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2017, pp. 4835–4839.

[277] Z. Tan, M. Mak, and B. K. Mak, "DNN-Based Score Calibration With Multitask Learning for Noise Robust Speaker Verification," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 26, no. 4, pp. 700–712, 2018.

[278] T.-P. Chen, L. Su *et al.*, "Functional harmony recognition of symbolic music data with multi-task recurrent neural networks." in *Proc. Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 2018, pp. 90–97.

[279] N. Tonami, K. Imoto, M. Niitsuma, R. Yamanishi, and Y. Yamashita, "Joint Analysis of Acoustic Events and Scenes Based on Multitask Learning," in *Proc. Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, 2019, pp. 338–342.

[280] Z. Wu, C. Valentini-Botinhao, O. Watts, and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Apr 2015, pp. 4460–4464.

[281] T. Kano, S. Sakti, and S. Nakamura, "End-to-End Speech Translation With Transcoding by Multi-Task Learning for Distant Lang. Pairs," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 28, pp. 1342–1355, 2020.

[282] S. Zhou, X. Zeng, Y. Zhou, A. Anastasopoulos, and G. Neubig, "Improving robustness of neural machine translation with multi-task learning," in *Proc. 4th Conf. on Machine Translation*, vol. 2, 2019, pp. 565–571.

[283] N. D. Lane, S. Bhattacharya, A. Mathur, P. Georgiev, C. Forlivesi, and F. Kawsar, "Squeezing Deep Learning into Mobile and Embedded Devices," *IEEE Pervasive Computing*, vol. 16, no. 3, pp. 82–88, 2017.

[284] T. Lee and A. Ndirango, "Generalization in multitask deep neural classifiers: a statistical physics approach," in *Proc. Advances in Neural Information Process. Syst.*, vol. 32, 2019, pp. 15 862–15 871.

[285] N. Zhuang, Y. Yan, S. Chen, and H. Wang, "Multi-task Learning of Cascaded CNN for Facial Attribute Classification," in *Proc. 24th Int. Conf. on Pattern Recognition (ICPR)*, 2018, pp. 2069–2074.

[286] Y. Gong, X. Luo, Y. Zhu, W. Ou, Z. Li, M. Zhu, K. Q. Zhu, L. Duan, and X. Chen, "Deep cascade multi-task learning for slot filling in online shopping assistant," in *Proc. AAAI Conf. on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6465–6472.

[287] D. Zhou and Q. He, "Cascaded Multi-Task Learning of Head Segmentation and Density Regression for RGBD Crowd Counting," *IEEE Access*, vol. 8, pp. 101 616–101 627, 2020.

[288] D. Fitzgerald, "Harmonic/percussive separation using median filtering," in *Proc. 13th Int. Conf. on Digital Audio Effects (DAFx'10)*, vol. 13, Graz, Austria, 2010.

[289] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," pp. 261–272, 2020.

[290] L. Van der Maaten and G. Hinton, "Visualizing High-Dimensional Data Using t-SNE," *J. of Machine Learning Res.*, vol. 9, pp. 2579–2605, Nov 2008.

[291] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2021, pp. 3875–3879.

[292] H. Li, K. Chen, and B. U. Seeber, "Auditory filterbanks benefit universal sound source separation," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, 2021, pp. 181–185.

[293] B. J. Borgström and M. S. Brandstein, "Speech Enhancement via Attention Masking Network (SEAMNET): An End-to-End System for Joint Suppression of Noise and Reverberation," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 29, pp. 515–526, 2021.

[294] H. Dinkel, S. Wang, X. Xu, M. Wu, and K. Yu, "Voice Activity Detection in the Wild: A Data-Driven Approach Using Teacher-Student Training," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 29, pp. 1542–1555, 2021.

[295] F. Chollet and Others, "Keras," https://keras.io, 2015. [Online]. Available: https://github.com/fchollet/keras

[296] K. N. Stevens, *Acoustic phonetics*. MIT Press, Cambridge, MA, USA, 2000, vol. 30.

[297] T. Dozat, "Incorporating nesterov momentum into adam," in *Proc. 4th Int. Conf. on Learning Representations (ICLR), Workshop Track*, 2016, pp. 1–4.

[298] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. of the North American chapter of the Assoc. for Computational Linguistics: Human Lang. Technologies*, 2016, pp. 1480–1489.

Appendix

The codes used for performing the various experiments in this thesis have been publicly shared. The links to access the different codes for each of the chapters are listed below:

Chapter 3 : `https://github.com/mrinmoy-iitg/Speech-Music-Classification-Using-SPT`

Chapter 4 : `https://github.com/mrinmoy-iitg/SMC_Phase_Features`

Chapter 5 : `https://github.com/mrinmoy-iitg/SM_HPSS_MTL`

Chapter 6 : `https://github.com/mrinmoy-iitg/MTGC_Speech_Music_Segmentation`

Moreover, the *Movie-MUSNOMIX* dataset created as part of the contribution of this thesis can be accessed at `https://github.com/mrinmoy-iitg/Movie-MUSNOMIX`.

List of Publications

1. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Speech/music classification using phase-based and magnitude-based features", in *Speech Communication*, vol. 142, pp. 34-48, 2022.

2. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Clean vs. Overlapped Speech-Music Detection Using Harmonic-Percussive Features and Multi-Task Learning," in *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, DOI: 10.1109/TASLP.2022.3164199, 2022.

3. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Speech/Music Classification Using Features From Spectral Peaks," in *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, vol. 28, pp. 1549-1559, 2020.

4. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Exploration of Speech and Music Information for Movie Genre Classification,". (submitted to IEEE Trans. on Affect. Comput., Jan 2023)

Conferences

1. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Foreground-Background Audio Separation using Spectral Peaks based Time-Frequency Masks," in *Proc. Int. Conf. on Signal Process. and Commun. (SPCOM)*, Bangalore, India, 2022.

2. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Detection of Speech Overlapped with Low-Energy Music using Pyknograms," in *Proc. Nat. Conf. on Commun. (NCC)*, Kanpur, India, 2021, pp. 1-6.

3. **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna and Prithwijit Guha, "Classification of Speech vs. Speech with Background Music," in *Proc. Int. Conf. on Signal Process. and Commun. (SPCOM)*, Bangalore, India, 2020, pp. 1-5.

Conferences (Other than thesis work)

1. Moakala Tzudir, **Mrinmoy Bhattacharjee**, Priankoo Sarmah, S. R. Mahadeva Prasanna, "Low-Resource Dialect Identification in Ao Using Noise Robust Mean Hilbert Envelope Coefficients," in *Proc. Nat. Conf. on Commun. (NCC)*, Mumbai, India, 2022.

2. Shikha Baghel, **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna, and Prithwijit Guha. "Automatic Detection of Shouted Speech Segments in Indian News Debates," in *Proc. Interspeech*, Brno, Czech Republic, 2021, pp. 4179-4183.

3. Shikha Baghel, **Mrinmoy Bhattacharjee**, S. R. Mahadeva Prasanna, and Prithwijit Guha. "Shouted and Normal Speech Classification Using 1D CNN." in *Proc. Int. Conf. on Pattern Recognit. and Mach. Intell. (PReMI)*, Tezpur, India, 2019, pp. 472-480.