Novel Methods for Time-Event Detection and Source Separation

A THESIS

submitted by

JILT SEBASTIAN

for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

JULY 2019

THESIS CERTIFICATE

This is to certify that the thesis titled, **"Novel Methods for Time-Event Detection and Source Separation"**, submitted by **Jilt Sebastian**, to the Indian Institute of Technology, Madras, for the award of the degree of **Doctor of Philosophy**, is a bona fide record of the research work done by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Hema A. Murthy
Research Guide
Professor
Dept. of Computer Science and Engineering
IIT-Madras, 600 036

Place: Chennai

Date: 04 July, 2019

ACKNOWLEDGEMENTS

I joined IIT Madras after resigning from a two-year old job to pursue my dream. I was interested in signal processing and machine learning, with no idea about research, how to do research in collaboration, what is paper writing and with a very little knowledge in computer science. A switch from electronics and communication discipline was smooth enough, even when there was a comprehensive exam which tests your understanding of core computer science subjects.

The entire credit goes to my guide and mentor **Prof. Hema A. Murthy** for all her efforts in moulding me towards the Ph.D. I would like to express my heart-felt gratitude to my beloved guide for all her efforts to provide a wonderful research experience. I thank her for her constant support and guidance during this period. She spent considerable amount of time helping me to understand all the concepts required for the research. She is very passionate about research and always motivated to do it with a helpful mind and strong commitment. She was instrumental in establishing research collaborations with various universities across the globe, which were very helpful in my research. During my research studies, I could spent valuable time with eminent professors in state-of-the-art research facilities with her constant aid and motivations. I honestly couldn't have asked for more.

I would also like to thank all the research collaborators. My sincere thanks to **Prof. Mriganka Sur** from department of brain and cognitive sciences of Massachusetts Institute of Technology, for inviting me for a research visit and enabling the collaboration in Computational Brain Research Project. I would like to thank **Prof. Shrikanth Narayanan**, for inviting me for a short research visit and for providing the financial support during the visit. He was always positive and motivated and wanted to enrich my research career. I was very fortunate to work under **Prof. Mathew Magimai.-Doss** during my tenure at Idiap Research Institute for one year, where I spent my time in understanding the direct-modelling techniques. He was very helpful with his timely-inputs and shaped up my independent research skills. I am thank full to him and the Swiss Government Excellence Scholarship Project for enabling this wonderful collab-

oration.

I will always remember the ever-motivating role of my doctoral committee members Prof. C. Chandra shekhar, Prof. R. Aravind, Prof. Kamakoti V, Prof. Deepak Khemani and, Dr. Sutanu Chakroborthi. Their valuable suggestions and feedback during the progress review meetings were instrumental in paving my research direction. I would like to thank Heads of the Department Prof. P. Sreenivasa Kumar and, Prof. Krishna Moorthy Sivalingam for providing excellent lab facilities to carry out the research work in the department. I am thankful to the administrative staff from CS office for their timely help. I also thank members of Sur Lab, SAIL lab, USC Catholic community, and Idiap for being very helpful and collaborative. I also thank all the staff members in the Department of computer science and engineering, IIT Madras for their assistance and cooperation.

The research group of speech technology in the institute, combining our Speech and Music Technology Lab (SMTL) and speech communications lab from the electrical department, was helpful in improving my knowledge in my research field. All the members (present and past) of the SMT Lab and DON Lab were helpful whenever I need any assistance and provided the kind of research environment one would seek to have. All of them were amazing colleagues. Life would have been so boring without these DONs, right from coffee/tea companionship to never-ending discussions. Many thanks to Rajeev, Shrey, Shreya, Manoj, Jom, Krishnaraj, Arun, Karthik, Nauman, Pavan, Manish, Lakshmanan, Akshay, Raghav, Jeena, Anusha, Lakshmi, Anju, Gayathri, Mari, Hari, Mahesh, Nathiya and all those who were part of my research life at sometime or the other.

The happy times spent with friends are never-ending. I am grateful to my friends Shyam, Lal, Basith, Moncy, Ullas, Anoop, Vishnu, Manas, Sudarsun, Priyathosh, Chandu and all football-mates. I am thankful to Fr. Jino, and members of Genesis community at IIT Madras and to my friends outside the institute Arjun, Paul, Jithu, Rickson, Naveen, Rony, Meby, Amal, Bibin, Colleagues at Accenture and many others for their constant support. I was fortunate to have great teachers both at school-level and at Amal Jyothi Engineering College who shaped my interests.

My sincere thanks to my parents, who were always by my side for achieving this dream. Special thanks to my brothers Jobit and Jesto and my in-laws for their interest and constant support. I could not have asked for more from my wife Divya, who was

always encouraging and supportive in all situations. I thank GOD almighty for all the blessings He has showered upon me. It was really fortunate to be a part of this beautiful campus. I plead forgiveness from anyone whose contributions I forgot to acknowledge; your wishes would remain a part of me forever. Thanks and best wishes to everyone.

Jilt Sebastian

04 July 2019

ABSTRACT

KEYWORDS: time-event detection, source separation, group delay, pitch estimation, onset detection, spike estimation, raw-waveform methods, signal analysis, signal-to-signal neural networks

Time varying signals contain information at various temporal resolutions. Extracting relevant time-related information and instances of importance from the signal is referred to as time-event detection (TED). Extraction of the signal of interest from a mixed signal is referred to as source separation. TED and source separation are important information retrieval tasks that require intelligent use of domain knowledge in signal processing and (or) machine learning.

Signal entities such as magnitude, time and, frequency are often employed for extracting important time-events. However, relative change in phase-based representations with respect to the instances of importance has not been studied extensively. The relevance of a well-known phase-based feature, known as group delay, is analysed in this thesis for extracting various information from signals. The thesis discusses the TED tasks from speech, music and neuronal signals and source separation/signal extraction from audio signals.

The property of group delay (GD) function relevant for time-event detection tasks is first analysed. This includes analyses of high resolution property of GD function for single pole and multi-pole systems. The fact that two closely-spaced pole locations are better resolved in group delay domain compared to spectral magnitude domain is discussed. This finding is then corroborated by TED tasks from speech, music and neuronal signals.

Pitch estimation from speech signals is considered which benefit from both the high-resolution property of GD, and the ability of grating compression transform (GCT) to capture the pitch-dynamics. The relevance of GD as a TED feature is further enriched by its adoption to musical percussive onset detection. The applicability of GD function is explored beyond the audio domain by considering spike estimation task. The

temporal positions of spikes are estimated from the calcium fluorescence signal by representing the input in the GD domain which is having a high resolution. The feasibility of end-to-end machine learning approach is then explored for this task. The ability of proposed signal-to-signal conversion neural network (S2S) to efficiently learn the task-specific features is further analysed and compared with the GD based approach.

The discriminative ability of the GD can also be exploited for signal extraction tasks. GD is used as a feature for efficiently learning time-frequency mask specific to the target source in singing voice separation task and for separating the melodic components in the musical mixture. The separation framework used for this task is based on recurrent neural network (RNN). This network is also employed as a separation stage for other signal extraction tasks such as percussive separation and speech denoising. This results in building hybrid systems for percussive onset detection from musical mixtures and gender identification under noisy conditions. Further, a task-dependant learning framework is explored for singing voice separation task using an adapted S2S network.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AGR	Automatic Gender Recognition
AMDF	Average Magnitude Difference Function
AP	Action Potential
ASR	Automatic Speech Recognition
AUC	Area Under ROC
AWGN	Additive White Gaussian Noise
BSS	Blind Source Separation
CASA	Computational Auditory Scene Analysis
CNN	Convolutional Neural Network
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
ESPS	Entropic Speech Processing System
FPE	Fine Pitch Error
GCT	Grating Compression Transform
GD	Group Delay
GECI	Genetically Encoded Calcium Indicators
GPE	Gross Pitch Error
HMM	Hidden Markov Model
HR	High Resolution
HPS	Harmonic/Percussive Separation
HPSS	Harmonic and Percussive Sound Separation
ICA	Independent Component Analysis
IRM	Ideal Ratio Mask
LPC	Linear Predictive Coefficients
LSTM	Long-Short Term Memory
MFDP	Mel-Frequency Delta-Phase
MFCC	Mel-Frequency Cepstral Coefficients

MIR	Music Information Retrieval
MODGD	Modified Group Delay
MODGDF	Modified Group Delay Feature
MSE	Mean Squared Error
NCCF	Normalised Cross Correlation Function
NMF	Non-negative Matrix Factorisation
OGB	Oregon Green Bapta
pYIN	Probabilistic YIN
RAPT	Robust Algorithm for Pitch Tracking
RIR	Room Impulse Response
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
SACF	Summary Auto-Correlation Function
SAR	Signalto Artifacts Ratio
SDE	Standard Deviation of Error
SDR	Signal to Distortion Ratio
SIR	Signal to Interference Ratio
SNR	Signal to Noise Ratio
SOS	Second Order Statistics
STE	Short-Time Energy
STFT	Short-Time Fourier Transform
STM	Spike Triggered Mixture
SVM	Support Vector Machine
S2S	Signal-to-Signal Conversion
TED	Time-Event Detection
TFR	Time-Frequency Representation
TMR	Target-to-Masker Ratio
UAR	Un-weighted Average Recall
VAD	Voice Activity Detection

NOTATION

σ_e	Standard deviation of pitch detection
α,γ	Modified group delay parameters
ω	Angular frequency
γ	regularisation hyper-parameter for RNN
θ	angle in anti-clockwise direction
τ	Group delay
I_g	Information Gain
$D(A\|\|B)$	KL divergence between A and B
$winscale$	Window Scale Factor
$e^{-\sigma}$	bandwidth of the system
τ_p	Modified group delay
τ_m	Group delay derived from Fourier transform magnitude
f_0	Fundamental frequency
ρ	Auto-correlation coefficient
z	z-transform
F_s	Sampling Frequency
$tlen, hlen$	Hamming window parameters

CHAPTER 1

Introduction

Analyses of signals at different temporal resolutions have given different levels of insights from an information theoretic perspective. Instances of significance vary depending upon the domain and nature of the signal. For example, from the perspective of music information retrieval (MIR), time-events such as a beginning of musical segment consisting of only a specific source, a metrical cycle and, an onset location are important at varied levels of time-resolutions. Detecting time events is often formulated as a peak estimation problem. The estimation is beneficial for higher level retrieval tasks. For example, musical note onset detection can be used for cover song detection.

Extracting "the signal of importance" is generally a harder task than estimating specific instances of importance. The performance depends on many factors such as the nature and type of signals, the kind of mixing process, and the number of sources present. Separating the speech of a target speaker from the overlapped speech, extracting the singing voice from musical mixture are examples of signal extraction techniques.

Information extraction from non-stationary signals traditionally employs amplitude, time, and frequency components. The benefits of using phase entity for these tasks have not been studied well in the literature. In fact, the phase-based methods are getting traction in the recent decades for various signal-processing tasks. The phase-aware signal processing in tandem with machine learning provides efficient solutions to research problems. Several phase-based features such as group delay and its variants, phase-difference, linear prediction phase, and instantaneous frequency have been used in the past for various speech processing applications. Negative frequency derivative of phase, known as group delay (GD), is one of the major phase-based features explored in the literature for various tasks in the speech community[1].

Information extraction is solely dependent on the specific task. The features capable of extracting relevant information from the speech signal may not be applicable to other domains. Some features such as Mel Frequency Cepstral Coefficients (MFCC) found

[1]The terms group delay and GD are used interchangeably in this thesis

to be useful for more than one application. Until recently most efforts have developed hand engineered features for extracting relevant information. Latest efforts on end-to-end information extraction using neural networks are getting to be topical. This thesis studies some aspects of information extraction using these purely data-driven methods in addition to the phase-based approaches.

1.1 Motivation

Obtaining the temporal positions of importance from a signal and extracting a complete signal from a mixed signal are very important in human-machine and machine-machine interactions. Estimating time-events are specific to the domain of application and need to be performed efficiently in real time. Some of the time-events might require a frequency-based analysis for accurate estimation whereas some might be based on temporal structure. Prediction and tracking of time-events are important pre-processing steps in many of the signal processing applications. Most of the real-time systems face challenges owing to the presence of background noise or other signals. Removing such interference is crucial in building efficient signal extraction systems.

1.1.1 Group Delay Analysis

Despite the success of group delay based approaches, a comprehensive analysis which considers the high-resolution property of group delay is missing in the signal processing research. The analysis of group delay functions presented in this thesis are motivated by earlier efforts towards examining the high-resolution property and its applications to various segmentation and feature extraction tasks in speech and musical signal processing. Earlier efforts have shown mathematically the importance of group delay functions in the context of single pole systems, we extend this to multi-pole systems. Most applications of GD in the literature are primarily for speech and music signals. This thesis extends their application to neuronal signals.

1.1.2 Time-Event Detection Tasks

Analysis of signals at different temporal levels are motivated by signals of different characteristics in various domains. Time-event detection methods proposed in this the-

sis are inspired by the success of group delay functions for segmentation applications in the past. This thesis looks at various domains and temporal resolutions for obtaining the time-events. For pitch estimation task, the relative change in pitch with respect to time can be considered as a time-event, and the typical pitch changes will be in milliseconds range. For musical signals, the percussive onsets can vary from milliseconds to several seconds owing to the change in time gap between the successive percussion strokes. This thesis extends the applicability of GD to neuronal signals in which action potentials are extracted from a slowly varying fluorescence signal. Multiple spikes can occur at the same time-resolution of a fluorescence signal as the fluorescence signal is sampled at a lower rate than the actual spike occurrence. The time gap between the successive spikes can be as long as several minutes. Application of group delay to these tasks are motivated by its ability to resolve closely spaced peaks and its non-model based representation. We further analyse the task-dependent feature learning approach for spike estimation task inspired by the success of sequence-to-sequence models.

1.1.3 Source Separation Tasks

High-resolution property of group delay can be utilised for applications which require the machine learning model to learn discriminative information between the source signals when they are mixed together. A modified version of group delay is used in the literature as a feature for performing speaker recognition and verification. This was the key motivation behind using this feature for signal separation tasks. The proposed methods using group delay points to its significance and use in machine learning methodologies for various signal processing-based applications. The possibility of using group delay in hybrid system is explored. Hybrid systems are the approaches in which individual stages perform specific sub-functions of the main task, instead of a single system performing the whole task. The thesis further discusses end-to-end models for performing the musical source separation task, motivated by its success in other speech applications.

1.2 Objective of the Thesis

This thesis aims to look at novel methods for time-event detection and signal extraction tasks, mainly using group delay based features. The major focus of the thesis is to analyse the high-resolution property of group delay functions and exploit it for applications in speech, music, and neuronal signals.

Figure 1.1 shows the general schematic of the time-event estimation and signal extraction tasks. Both the signal processing and machine learning solutions to these two applications are fitted in a single framework.

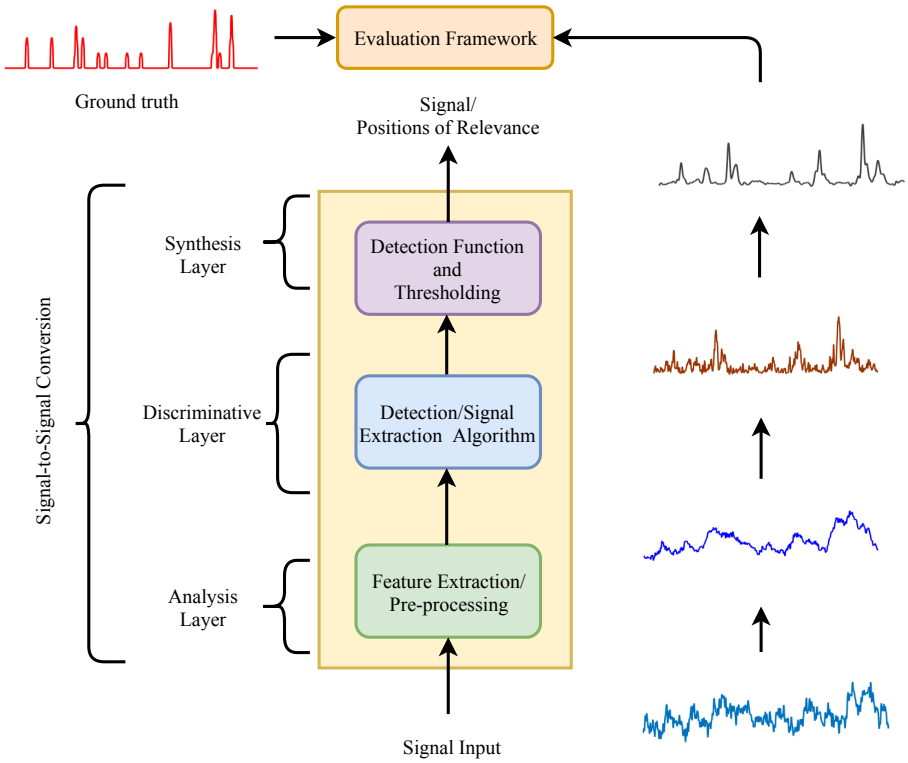


Figure 1.1: Schematic of time-events/signal extraction applications.

The time-event detection problem is formulated as a peak extraction problem in the group delay domain. A pre-processing is often employed to enhance the desired characteristics in the input signal (or to annihilate the undesired ones). The "modified" signal is then used as the input for the detection algorithm based on group delay. This computation results in representing the enhanced signal in the GD domain, exploiting its high-resolution property. The discriminatory information is emphasised in the GD domain. In the final stage, the temporal instances of importance are extracted by converting the GD-based output to a detection function and then extracting the tem-

poral locations using a threshold. This post-processing step detects the positions of relevance. Each stages of the peak estimation problem can be modelled using neural network layer(s). This signal-to-signal (S2S) neural network jointly models each of these stages by means of an analysis and synthesis layer and a time-distributed multi-layer perceptron (MLP) sandwiched between them. The conformity between S2S and signal processing solution is shown in the schematic diagram (Figure 1.1 (*left*)).

Signal extraction applications such as singing voice separation and speech denoising use neural network-based approaches for obtaining the target sources. Traditional methods use spectral magnitude based features for obtaining a time-frequency (T-F) mask specific for the source. This problem is approached by using a group delay-based feature as the input to the network instead of using spectrum-based features. Feature extraction (pre-processing) is followed by a T-F mask estimation using GD corresponding to the target source. This is mentioned as a signal extraction algorithm in Figure 1.1. The estimated target spectrum is considered as the detection function, from which the target audio source is then reconstructed. Stages of this network can also be interpreted in terms of S2S which directly learn the mapping from source to target audio. It has analysis and synthesis layers in addition to the neural network-based mask estimation. Pre-processing is done using a set of convolutional layers to get the magnitude-specific information, which is fed to the hidden layers to extract the signal of importance. The source signals are reconstructed with the help of set-aside phase representation.

1.3 Organisation of the Thesis

The analysis of the group delay functions and its applications to various tasks is a significant component of this thesis. Chapter 2 provides an in-depth literature survey of group delay functions for various tasks. Chapters with a uniform format follow the second chapter. In each of them, the general idea of the chapter is first discussed followed by the literature review of the considered task. This is then succeeded by the proposed approach(es) and their evaluations.

- Chapter 2 provides an overview of group delay functions and their properties. Two variants of group delay and their importance in speech and audio signal processing is discussed. Major applications of group delay in the speech and music literature are reviewed towards the end of the Chapter 2.
- Chapter 3 discusses the theoretical analysis of high resolution property of the group delay (GD) functions. The high resolution property and its related works

are reviewed. The proof for the high-resolution property is presented for the single and multi-pole systems. The theoretical analysis is followed by numerical analyses towards the end of the chapter.

- Chapter 4 presents the time-event detection (TED) tasks. The pitch estimation from speech, percussive onset detection from music, and spike estimation from neuronal signals are discussed. Modified-group delay-based representation is employed in conjunction with grating compression transform (GCT) to obtain the pitch estimates. Musical time-events are obtained at stroke-level from percussive instruments using minimum-phase GD. The final part of the chapter focuses on group delay-based spike estimation. The spike estimation task is more comprehensively explained, since the GD is applied beyond audio domain for this task. It also presents an end-to-end machine learning alternative to spike estimation task using signal-to-signal neural networks. The chapter ends with comparison of the proposed approaches.

- Chapter 5 discusses the source separation task. A discussion on recent deep-learning-based algorithms follows the literature on conventionally used features for this task. Modified group delay is then presented as a proposed feature for musical source separation. Signal-to-signal network is then explored for the separation of singing voice. Hybrid systems for percussive onset detection and automatic gender identification from mixed signals are presented towards the end of this chapter. These approaches make use of source separation-based preprocessing stages.

- Chapter 6 provides a summary of the approaches presented, and the general conclusion of the thesis. It also discusses the criticisms and the possible future directions.

1.4 Major Contributions of the Thesis

The major contributions of the thesis are:

- High resolution property of GD functions in the context of multi-pole systems. This proof validates the usefulness of group delay for various applications proposed in the literature.
- A pitch estimation algorithm which benefits from the high-resolution property of group delay functions and the ability of the grating compression transform (GCT) to track pitch dynamics.
- Novel signal processing based approach to onset detection for percussion instruments.
- Spike information estimation from neuronal signals using group delay based signal processing and end-to-end neural network model.
- Musical source separation using modified group delay-gram as a feature to learn the spectral mask in a state-of-the-art RNN system. Purely data-driven end-to-end model for singing voice separation task.
- Hybrid systems for percussive onset detection from musical mixture, and language-independent gender identification under noisy and low-resource conditions.

CHAPTER 2

Theory of Group Delay Functions and its Applications

2.1 Introduction

Signal processing applications use diverse entities such as magnitude, frequency, phase and time to perform the desired task. Phase based representations are replaced by the group delay function, avoiding the task of unwrapping the phase. Practical signals consist of various sinusoidal components. Group delay refers to the delay in time for the amplitude envelopes of each of these elements. It is a frequency-dependent measure because various frequency components have different delays when passed through a non-linear phase system.

This chapter presents an overview of group delay function which is used as a representation for various TED tasks in Chapter 4, and source separation in Chapter 5. The theory of group delay processing and its properties are explained in the first section (Section 2.2). When zeroes are close to the unit circle in the z-domain, the group delay function misbehaves. Various signal processing methods have been proposed in the literature to alleviate this problem. These techniques are discussed in Section 2.3. Applications of group delay in speech and audio processing are then reviewed in Section 2.4. Section 2.5 discusses the conclusions of this chapter.

2.2 Group Delay Functions

The information is encoded in both magnitude and phase in the spectral representation of signals. Importance of the phase spectrum has been established only in the last few decades. In automatic speech recognition, the random phase values altered the recognition rates significantly compared to the actual and reconstructed phase values (Shi *et al.*, 2006) under various signal-to-noise ratios (SNRs). The relevance of phase-based processing is exploited for speaker and speech recognition applications in (Rajesh M. Hegde, 2005; Bozkurt *et al.*, 2007). The group delay function has been proposed

as an alternative to magnitude spectrum for segmentation and feature extraction tasks in speech processing (Nagarajan *et al.*, 2001; K. *et al.*, 2004; Lakshmi and Murthy, 2008; Rasipuram *et al.*, 2008*b*; Kumar and Murthy, 2009; Janakiraman *et al.*, 2010; Murthy and Yegnanarayana, 2011; Bellur and Murthy, 2013*a*; Golda and Murthy, 2013; Shanmugam and Murthy, 2014*b*).

Group delay (GD) also has the same information as in the Fourier transform magnitude. Let $x[n]$ be a discrete time signal, and let $X(e^{j\omega})$ be its discrete time Fourier transform (DTFT). Group delay ($\tau(e^{j\omega})$) is defined as the negative derivative of the unwrapped phase spectrum with respect to frequency.

$$\tau(e^{j\omega}) = -\frac{d\{\arg(X(e^{j\omega}))\}}{d\omega}. \tag{2.1}$$

where, $\arg(X(e^{j\omega}))$ is the Fourier transform phase and, ω is the angular frequency of the signal. The phase and the group delay spectrum are shown for a two-pole system in Figure 2.1. Observe that it is difficult to interpret the phase function, but not the group delay representation in which the peaks implicitly correspond to the pole locations.

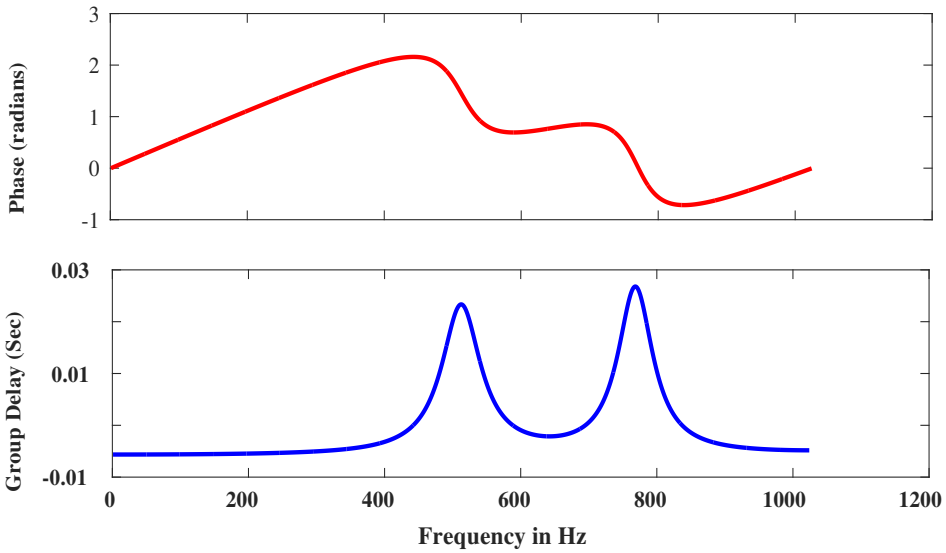


Figure 2.1: Fourier transform phase (*top panel*), and group delay (*bottom panel*) for a minimum phase two-pole system.

The group delay function can be computed directly from the signal (Yegnanarayana *et al.*, 1984; Oppenheim and Schafer, 1990) and its time shifted version.

Considering the Fourier representation for a discrete time signal $x[n]$:

$$X(e^{j\omega}) = |X(e^{j\omega})|e^{j\arg(X(e^{j\omega}))} \tag{2.2}$$

$$\log\ X(e^{j\omega}) = \log\ (|X(e^{j\omega})|) + j\{\arg(X(e^{j\omega}))\} \tag{2.3}$$

$$\arg(X(e^{j\omega})) = Im[\log\ X(e^{j\omega})]. \tag{2.4}$$

Hence, phase representation $\arg(X(e^{j\omega}))$ is the imaginary part of the log of the complex spectrum.

Now, computing group delay using Equation 2.1:

$$\tau(e^{j\omega}) = -\frac{d(Im(\log(X(e^{j\omega}))))}{d\omega} \tag{2.5}$$

$$\tau(e^{j\omega}) = \frac{X_R(e^{j\omega})Y_R(e^{j\omega}) + Y_I(e^{j\omega})X_I(e^{j\omega})}{|X(e^{j\omega})|^2} \tag{2.6}$$

where $X(e^{j\omega})$ and $Y(e^{j\omega})$ are the Fourier transforms of $x[n]$ and $nx[n]$ respectively, and the real and imaginary parts are denoted by R and I respectively.

This straight-forward computation bypasses unwrapping task in group delay-based applications. However, the denominator $|X(e^{j\omega})|^2$ causes ill-behaviour for group delay for non-minimum phase signals (Murthy, 1991). Whenever the value of the magnitude spectrum becomes zero, the GD give rise to spurious peaks as the denominator becomes zero in Equation 2.6. This is illustrated in Figure 2.3 (*middle panel*). Observe the noise-like behaviour of group delay when the zeros are on the unit circle (non-minimum phase). Hence for practical signals, the signal is either converted to its minimum phase variant to perform group delay analysis (Yegnanarayana and Murthy, 1992; Shanmugam and Murthy, 2014*a*), or compute a modified version of the group delay ($\tau_p(e^{j\omega})$) which directly considers the non-minimum phase signals (Murthy, 1991; Rajesh M. Hegde, 2005). These variants are explained in detail in Section 2.3.

Alternative representations of group delay widen its interpretability. Group delay can also be obtained as the Fourier cosine transform of the weighted cepstrum (Yegnanarayana *et al.*, 1984; Oppenheim and Schafer, 1990; Murthy and Yegnanarayana, 2011). This is because both the log magnitude and the phase spectra can be represented in terms of the cepstral coefficients. The log magnitude spectrum is determined from cepstral representation (Yegnanarayana *et al.*, 1984) as:

$$ln \mid X(e^{j\omega}) \mid = c_1[0]/2 + \sum_{n=1}^{\infty} nc_1[n]\cos[n\omega] \tag{2.7}$$

where $c_1[n]$ refers to nth cepstral coefficient. The unwrapped phase is computed as follows

$$\arg(X(e^{j\omega})) = -\sum_{n=1}^{\infty} nc_2[n]\sin[n\omega] \tag{2.8}$$

where $c_1[.]$ are the cepstral coefficients of the minimum phase signal derived from the spectral magnitude and $c_2[.]$ are that derived from the spectral phase. For minimum phase signals, c_1 and c_2 are the same ($c_1 = c_2 = c$). Hence, a group delay function can be derived from the set of cepstral coefficients as,

$$\tau(e^{j\omega}) = -\sum_{n=1}^{\infty} nc[n]\cos[n\omega] \tag{2.9}$$

The phase (Equation 2.8) and the spectral magnitude (Equation 2.7) are related via cepstral coefficients and the group delay function is the Fourier transform of the weighted cepstrum (Equation 2.9).

As a representation which has similar spectral characteristics to that of the magnitude spectrum, group delay function is of interest to the research community. The relevant properties which distinguish the magnitude spectrum and the group delay function are discussed in the following subsections. The group delay function obtained from the Fourier transform magnitude is denoted by $\tau_m(e^{j\omega})$ (Equation 2.6), and the minimum phase equivalent signal derived from the Fourier transform magnitude is represented by and $\tau_p(e^{j\omega})$, respectively.

2.2.1 Properties of group delay functions

Basic properties of group delay consider its dependence on minimum phase criteria.

- Roots of the system transfer function is manifested as peaks or valleys in the GD domain, depending on the poles or zeros, respectively.
- High resolution and additive properties.
- For a minimum phase signal

$$\tau_p(e^{j\omega}) = \tau_m(e^{j\omega}) \tag{2.10}$$

- For maximum phase signal

$$\tau_p(e^{j\omega}) = -\tau_m(e^{j\omega}) \tag{2.11}$$

- For a mixed phase signal

$$\tau_p(e^{j\omega}) \neq \tau_m(e^{j\omega}) \tag{2.12}$$

- For a root location on the unit circle,

$$\tau_p(e^{j\omega}) = \tau_m(e^{j\omega}) = \infty \tag{2.13}$$

Characteristics of group delay are exhaustively covered in (Murthy and Yegnanarayana, 2011). Two significant properties of group delay which makes it suitable for signal processing applications are reviewed here. These properties are utilised in this thesis for TED and source separation tasks.

2.2.1.1. Additive property

Convolution in time becomes multiplication in the frequency domain and becomes an addition in the phase domain representation. Hence, a cascade connection of resonators is indicated as their summation in the group delay domain. This is termed as additive property. This property is common for any phase domain representation inclusive of GD.

Considering a linear time-invariant (LTI) system formed by a cascade connection of N resonators whose impulse responses are $h_1[n], h_2[n], \ldots, h_N[n]$ respectively. Impulse response of the overall system is,

$$h[n] = h_1[n] * h_2[n] * ... * h_N[n] \tag{2.14}$$

where (*) is the convolution operation. By convolution theorem, the frequency response of this system is the product of responses of its constituent resonators.

$$H(e^{j\omega}) = \prod_{i=1}^{N} H_i(e^{j\omega}) \tag{2.15}$$

where $H_i(e^{j\omega})$ refers to the response of system i.

Converting the responses to polar form,

$$H(e^{j\omega}) = \prod_{i=1}^{N} |H_i(e^{j\omega})| arg(H_i(e^{j\omega})) = \prod_{i=1}^{N} |H_i(e^{j\omega})| \sum_{i=1}^{N} arg(H_i(e^{j\omega})) \tag{2.16}$$

Summation term in Equation 2.16 is the Fourier transform phase of the combined system. Group delay function ($\tau(e^{j\omega})$) can be now computed as,

$$\tau(e^{j\omega}) = -\sum_{i=1}^{N} \frac{\partial(arg(H_i(e^{j\omega})))}{\partial\omega} \tag{2.17}$$

$$= \sum_{i=1}^{N} \tau_{h_i}(e^{j\omega}) \tag{2.18}$$

where, τ_{h_i} is the group delay response for the ith resonator.

Figure 2.2 shows the additive property of group delay for a system consisting of two poles. The signal is obtained by exciting a single pole system with an impulse. Convolution between them in the time-domain leads to the output $y[n]$. The response in group delay domain is additive, and that in magnitude spectrum is multiplicative. Observe that the magnitude spectrum of the system is less discriminative compared to group delay spectrum, which we will discuss next.

2.2.1.1. High resolution property

The n dB bandwidth of group delay function is always lesser than that of the magnitude spectrum for minimum phase signals (Sebastian *et al.*, 2016). This means that the group delay function is sharper than the magnitude spectrum. This property is known as the high-resolution property and is true for both single and multi-pole systems. High-resolution property enables group delay to resolve closely-spaced peaks in the frequency domain.

The ability of the group delay functions for resolving formants which are closely spaced in the spectrum is illustrated in (Murthy and Yegnanarayana, 1991). High resolution property of group delay is shown mathematically for single pole systems by considering the n dB bandwidth in (Kumar, 2015). This is examined in greater detail in Chapter 3 in this thesis. Each of the poles in Figure 2.2 exhibits high resolution property

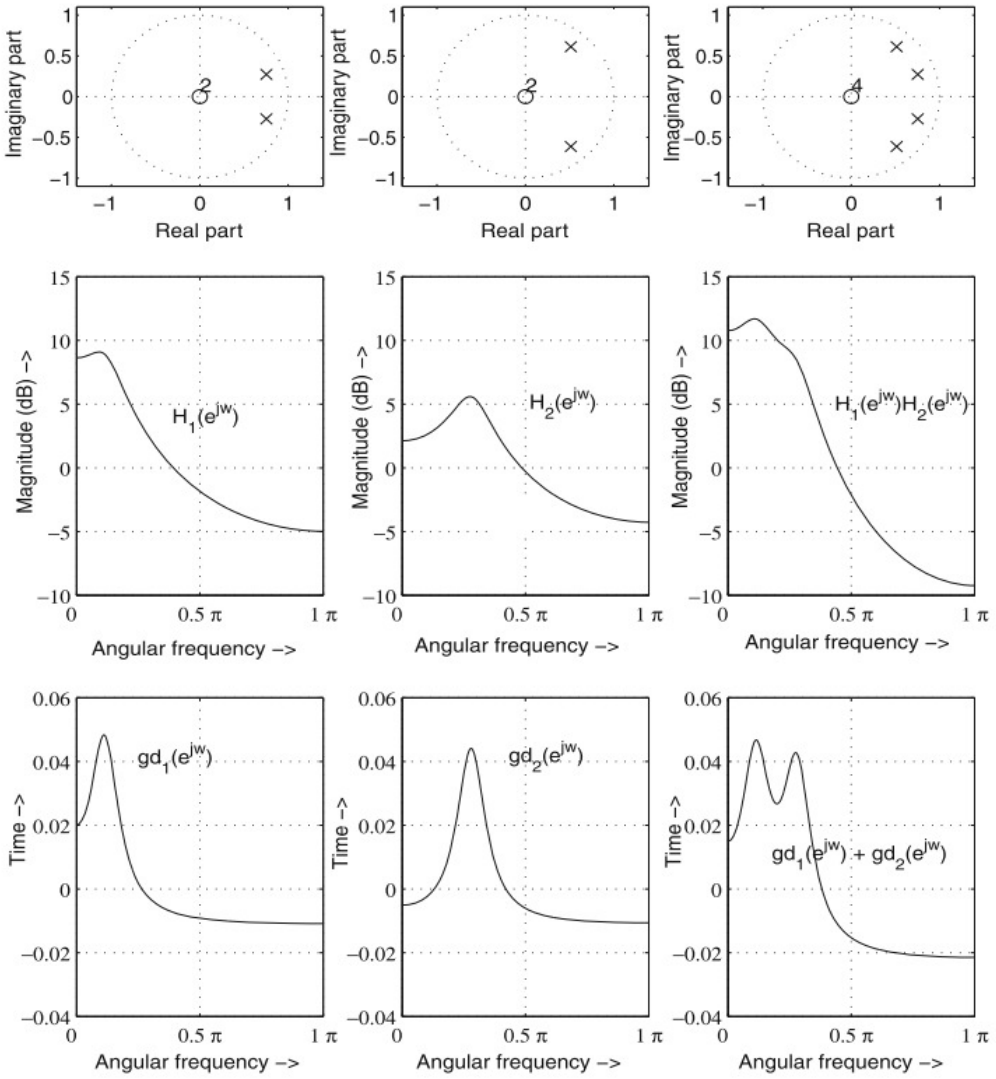


Figure 2.2: Additive property of the group delay functions. Z-transform representations of input signal ($x[n]$), system impulse response ($h[n]$), and the output ($y[n]$ where $y[n] = x[n] * h[n]$) are shown in the upper panel. Magnitude spectra of these signals are shown in the middle panel and the corresponding group delay functions in the bottom panel(Taken from (Murthy and Yegnanarayana, 2011)).

in addition to the additive property. Observe that each resonator is sharper in the GD domain compared to that of the magnitude spectrum.

Since the practical signals are not minimum phase, it is impossible to apply group delay processing directly. This is because zeros cause the denominator of Equation 2.6 to becomes zero as seen in Figure 2.3. Hence, two modifications are traditionally made to enable group delay processing.

2.3 Modifications to Group Delay

The uncertainty of the group delay representation for non-minimum phase signals limits its utility. There are two ways to alleviate this issue while handling practical signals. One solution is to convert the non-minimum phase signal to minimum phase equivalent (Nagarajan, 2004) and then perform minimum phase group delay analysis on this signal. Another solution is to define a modified version of group delay for non-minimum phase signals (Murthy, 1991). Both of the variants result in minimum-phase equivalent representations and are discussed in the following subsections.

2.3.1 Converting the signal to minimum phase

Conversion to minimum phase is of great significance since, both the peaks and valleys are better resolved only for the minimum phase signal in the group delay spectrum. Group delay processing can be performed after converting the signal to minimum phase.

The theory of minimum phase signals has been developed and used extensively in signal processing (Berkhout, A. J., 1974). If a signal and its inverse are one-sided and energy-bounded, they are termed as minimum phase signals. For these signals, all the poles and zeros lie within the unit circle in the Z-domain. It is possible to decompose a non-minimum signal in Z-domain to a minimum phase and an all-pass component. The need for this explicit decomposition is avoided in (Nagarajan *et al.*, 2001) by modifications to the signal such that the Z-domain representation has zeros and poles inside the unit circle.

For a root inside the unit circle at a distance 'x' from it, the magnitude spectrum will be the same as that of the root outside the unit circle at a distance '$\frac{1}{x}$', and with the same angular frequency. i.e., for any type of system, the magnitude spectrum will be the same if the angular position of poles and zeros are identical. The minimum phase property is proved in (Nagarajan *et al.*, 2001). The minimum-phase signal is obtained by taking the causal part of the signal computed by finding the inverse Fourier transform of the magnitude spectrum of the non-minimum phase signal. If the input signal is a positive function, it can also be considered as a magnitude spectrum by symmetrizing it. The roots of minimum phase signal obtained by this method lie inside the unit circle with the same angular frequency. This property is used in (Murthy, 1991) to derive a

minimum phase variant of the non-minimum phase signal by reflecting the roots inside the unit circle.

Algorithm for converting a non-minimum phase signal to minimum phase signal:

- Generate the non-minimum phase signal $x_{nmp}[n]$ by exciting a non-minimum phase system with an impulse response.
- Compute the Fourier transform magnitude $|x_{nmp}[\omega]|$ of the non-minimum phase signal.
- Take inverse Fourier transform of $|x_{nmp}[\omega]|$ to get the root cepstral representation $x_c[n]$.
- Take the causal portion of $x_c[n]$ to obtain the minimum phase equivalent of $x_{nmp}[n]$.

Analyses of the minimum-phase system based on proximity of the pole/zero locations to the unit circle are discussed in detail in (Nagarajan *et al.*, 2001).

2.3.2 Group delay for non-minimum phase signals

It has been shown in (Madhumurthy and Yegnanarayana, 1989) that group delay functions represent signal information for a minimum phase signal. For practical non-minimum phase signals, zeros close to the unit circle create artificial spikes in the group delay domain. These zeros are caused by various factors such as short-time analysis in audio signal processing and the glottal return phase in the speech signal and pitch trajectories. Hence, it is necessary to remove the effect of zeros on the group delay response.

A modified version of group delay is proposed in (Murthy, 1991) for non-minimum phase signals which eliminates spectral zeros and thereby providing a highly-resolved group delay spectrum. This is performed by replacing $|X(e^{j\omega})|$ in the denominator of Equation 2.6 by its smoothed envelope $|S(e^{j\omega})|$. This smoothing can either be cepstral-based or moving average-based. The importance of cepstral smoothing technique is described in (Rajesh M. Hegde, 2005). The Equation for modified group delay ($\tau_c(e^{j\omega})$) is given by,

$$\tau_c(e^{j\omega}) = \frac{X_R(e^{j\omega})Y_R(e^{j\omega}) + Y_I(e^{j\omega})X_I(e^{j\omega})}{|S(e^{j\omega})|^2} \tag{2.19}$$

After this modification, it was observed that the dynamic range was very large (owing to the approximate estimation of the envelope). Two new empirical parameters

α and γ are introduced to control the dynamic range. Hence, the equation is further modified (Murthy and Gadde, 2007) as,

$$\tau_p(e^{j\omega}) = (\frac{\tau_c(e^{j\omega})}{|\tau_c(e^{j\omega})|})|\tau_c(e^{j\omega})|^{\alpha}, \tag{2.20}$$

where,

$$\tau_c(e^{j\omega}) = \frac{X_R(e^{j\omega})Y_R(e^{j\omega}) + Y_I(e^{j\omega})X_I(e^{j\omega})}{|S(e^{j\omega})|^{2\gamma}}. \tag{2.21}$$

Modified group delay function is computed as follows:

- Let $x[n]$ be the time domain signal. Compute the DFT of $x[n]$ and $nx[n]$. These are denoted as $X[k]$ and $Y[k]$, respectively.
- Compute the modified group delay $\tau[k] = \frac{X_R[k]Y_R[k]+X_I[k]Y_I[k]}{|S[k]|^2}$, where $S[k]$ is the cepstrally smoothed magnitude spectrum.
- Apply the parameters α and γ (Typically $0 < \alpha \leq 1$ and $0 < \gamma \leq 1$ (Murthy, 1991; Rajesh M. Hegde, 2005)) to the above Equation to get the equivalent representations of Equations 2.21 and 2.20 respectively, in the discrete domain.

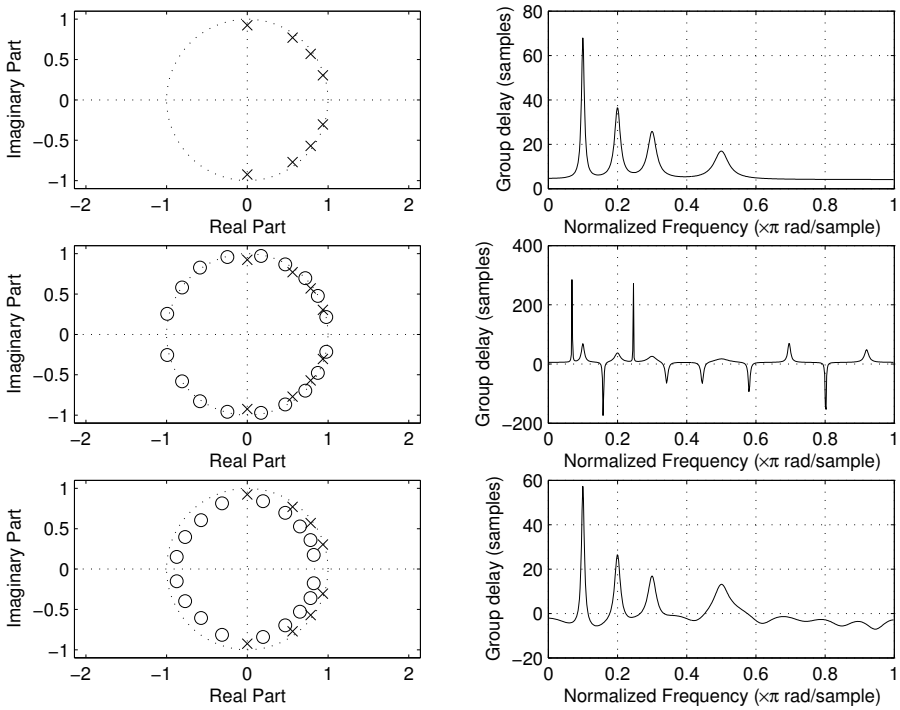


Figure 2.3: Significance of modified group delay representation for non-minimum phase signals. (*top panel*) A four pole system without zeros, (*middle panel*) system with zeros on the unit circle, and (*bottom panel*) with zeros pushed radially inside the unit circle with their group delay spectra.

Figure 2.3 shows the significance of modified group delay[1]. A four pole minimum phase system and its corresponding group delay representation is shown in the

[1]Taken from (Rajesh M. Hegde, 2005)

top panel. The pole locations are clearly emphasised in the GD domain. However, introduction of zeros to the unit circle (practical signals) causes zeros to the Fourier transform representation (Equation 2.6) and results in noisy behaviour of group delay. This is shown in the middle panel. Pushing of zeros inwards from the unit circle by computation of modified group delay results in retaining the shape of group delay spectrum similar to that of the minimum phase version.

2.3.2.1 Modified group delay feature (MODGDF)

The modified group delay (MODGD) resolves the peak locations better than that of the magnitude spectrum for a non-minimum phase signal. This enable the use of group delay as a signal processing tool and also as a feature for various processing tasks. Modified group delay is transformed into the cepstral domain using the discrete cosine transform (DCT). It acts as a data independent de-correlator (Yip and Rao, 1997). The cepstral features $c_{mod}[n]$ are obtained as,

$$c_{mod}[n] = \sum_{k=0}^{k=N_f} \tau_m[k] \cos(\frac{n(2k+1)\pi}{N_f}) \tag{2.22}$$

where N_f is the DFT order and $\tau_m[k]$ is the modified group delay spectrum. The number of such coefficients can be limited to the first k coefficients which accounts for most of the energy in the MODGD spectrum. This feature is known as Modified Group Delay Feature (MODGDF). The dimension of the MODGDF depends upon the order of the DFT. DCT is used to reduce the dimension of the feature. The velocity and acceleration values can be determined by computing the MODGDF across the time frames. They are also concatenated as additional dimensions in MODGDF similarly to that of MFCCs.

The complementary nature of MODGDF with respect to traditional acoustic features is discussed in (Rasipuram *et al.*, 2008*a*). The MFCC and spectral phase-based MODGDF are concatenated to form joint features. These features are used in (Rasipuram *et al.*, 2008*a*) for recognising phonetic unit classes, where performance improvement is observed compared to other fusion techniques. In (Kumar *et al.*, 2010), ASR using feature switching based on Kullback Leibler (KL) divergence is proposed.

2.3.2.2 MODGD-Source

Root cepstral smoothing (Murthy, 1994) is applied to power spectrum of a signal to obtain its flattened spectrum. Modified group delay is computed on this source spectrum to get the MODGD-Source. The peaks observed on the MODGD-source are sharper than that of the flattened spectrum. High resolution property of GD applies only to minimum-phase signals. Since the modified group delay is computed over a source signal which is non-minimum phase, the high resolution property does not hold. However, the peaks observed in the flattened spectrum (corresponds to the fundamental frequency and its harmonics for speech) are emphasized in MODGD-Source.

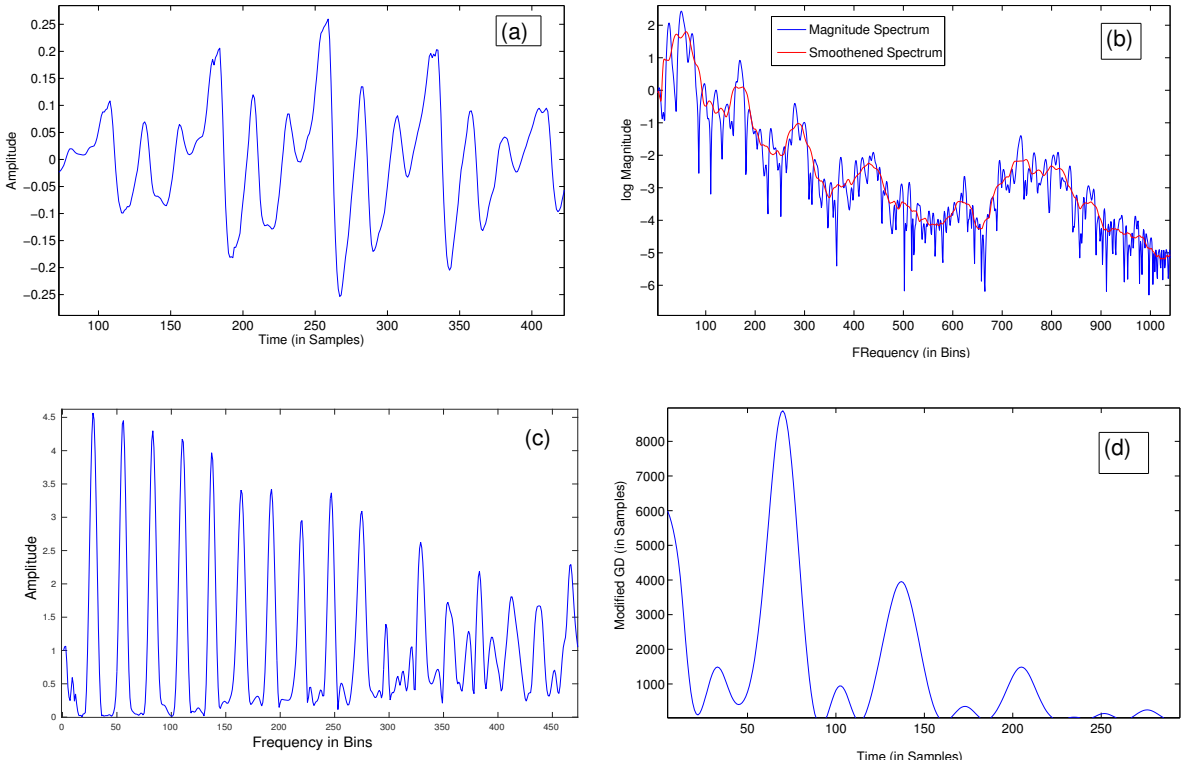


Figure 2.4: Algorithm: (a) A frame of speech, (b) log magnitude & smoothed spectrum of (a) superimposed over it, (c) flattened spectrum, and (d) MODGD-source of (a)).

For every frame of speech, the power spectrum is computed. It is then divided by its smoothed envelope, which represents the system characteristics. The flattened spectrum thus contains the source related information. Pitch and its harmonics are well-represented in the flattened spectrum. The modified group delay function computed for this signal is known as MODGD-Source. Peaks are observed at multiples of the pitch period when the group delay function is computed over the flattened spectrum. The flattening process is illustrated in Figure 2.4. The pitch and its harmonics are emphasised

in the GD domain (Figure 2.4 (d)). 2048 point FFT is taken to obtain the magnitude spectrum. To improve the visualization of flattened spectrum and its MODGD, the scales have been adjusted. This variant of modified group delay is exploited for pitch estimation tasks in Rajan (2017) and in this thesis.

2.4 Group Delay Applications in Speech and Music

Group delay functions have been used for several audio signal processing applications in the past. High-resolution and additive properties of GD is utilised for most of these applications. Group delay is used for applications such as pitch estimation (Yegnanarayana and Murthy, 1992), formant estimation, speech segmentation into syllable-like units (Nagarajan *et al.*, 2003), text-to-speech synthesis systems (Shanmugam and Murthy, 2014*a*), speech (Murthy and Gadde, 2007) recognition, speaker recognition and verification (R.M.Hegde *et al.*, 2004), language identification (Nagarajan and Murthy, 2006), voice activity detection, and emotion detection from speech signals. In all these tasks, pre-processing the speech signal using group delay/extraction of features from group delay have significantly improved the performance.

GD has been used recently for various Music Information Retrieval (MIR) applications. This includes tonic identification (Bellur *et al.*, 2012), melody mono-pitch and multi-pitch estimation (Rajan, 2017), percussive onset detection (Kumar *et al.*, 2015) and musical source separation (Sebastian and Murthy, 2016). Some of these applications are discussed in this section.

2.4.1 Pitch and formant estimation

Pitch estimation is the task of detecting and tracking the fundamental frequency of vibrations of vocal folds in a speech signal. Formants are resonances of the vocal tract and varies depending upon the vowels. Similar to pitch, the range of formants varies for each person, within the allowable range for the vowel under consideration. Pitch and formant estimation is of importance as a pre-processing step for many high-end applications such as text-to-speech synthesis, melody-based recommendation systems and speaker adaptive training in automatic speech recognition (ASR). Group delay has been used in the past for estimating both the vocal tract information (formants) and

the source information (pitch) from the speech signals. GD is primarily used in these applications as a peak enhancer to accurately detect the spectral peaks by exploiting its high-resolution and additive properties. MODGD-Source is used in (Murthy, 1991) for pitch estimation, and it is shown in (Yegnanarayana and Murthy, 1992) that it can be used for estimating the formant locations. The pitch estimation algorithm using GD is robust to noise and is superior to many of the existing algorithms. A method for estimation of number of speakers in a multi-pitch environment is developed in (Rajan, 2017) using MODGD-Source features. GD based formant estimation is first proposed in (Duncan *et al.*, 1989). Formant estimation using spectral root GD approach is presented in (Murthy and Yegnanarayana, 1991) which provides consistent formant estimates, as compared to traditional cepstral and linear prediction based approaches. A variant of this algorithm is explained in (Murthy and Yegnanarayana, 2011) to estimate the formant using homomorphic analysis.

Epoch is a source related information contained in the speech signal. It refers to the instance of significant excitation of the vocal tract system in speech production. An algorithm for detecting significant instances of excitation is devised based on group delay in (Rao *et al.*, 2007). Such instances are of two types; the epochs where glottal closure happens for the voiced speech and random bursts for the unvoiced speech. The approximate epoch locations are extracted in the first phase using the Hilbert transform of the LP residual of the speech signal. In the second phase, significant excitation instances are estimated using GD processing around the approximate epoch locations. A technique based on zero resonance frequency filtering is proposed in (Sri Rama Murty and Yegnanarayana, 2008) for epoch extraction. This method uses GD based processing on LP residual of the speech signal.

2.4.2 Speech and speaker recognition

Two major speech applications in which group delay has been explored are speech recognition and speaker identification. Novel segmentation and feature extraction stages are introduced using group-delay based approaches for speech recognition (Kumar and Murthy, 2009; Rasipuram *et al.*, 2008*b*; Padmanabhan and Murthy, 2010; Murthy and Yegnanarayana, 2011). In (Kumar and Murthy, 2009), the GD based segmentation algorithm is used to cut the speech signal at syllable-level. In (Rasipuram *et al.*, 2008*b*),

group delay based features are integrated into the linguistic search space. The complementary nature of the group delay features in comparison to magnitude based features is also discussed. Chirp group delay is used in (Bozkurt *et al.*, 2007) as a feature for automatic recognition of phonemes. Chirp group delay based feature is proposed for ASR with a comparable performance to MFCCs in (Jayesh and Ramalingam, 2016).

A method to automatically segment and label speech signals into syllable-like units is performed in the context of ASR in (Sarada *et al.*, 2009). GD based algorithm is used to get the pseudo-syllables in the first phase. An unsupervised technique is then used to group together similar segments. The clusters formed by this method is further used for acoustic modelling using an HMM framework. The of use MODGDF as a feature for ASR is illustrated with the help of extensive experiments in (Rajesh M. Hegde, 2005).

The effect of GD for speaker recognition is studied in (Thiruvaran *et al.*, 2007). Two feature representations based on GD are proposed: First one addresses the masking effect of spikes in GD domain using log compression and the second one uses a sub-band based approach to constrain the masking of certain bands. The work by (Dey *et al.*, 2011) explores the use of the feature-switching framework in speaker recognition task. It uses the fact that some features are better than the other in discriminating some specific classes. MODGDF is used with MFCC and LPCC in such a framework for speaker recognition and verification. All-pole model based GD features are used for speaker identification in (Rajan *et al.*, 2013). A comparable performance to that of magnitude based features is achieved using this method, and an improved performance is observed when the utterances with a high vocal effort are recognised. GD functions obtained from LP model are used in (Bastys *et al.*, 2010) as features for speaker identification.

Spoofing detection is performed using deep neural networks (DNN) in which the input feature is obtained from group delay functions. A similar approach is employed in (Diment *et al.*, 2016) for environmental sound event recognition in which GD derived from all pole models is used as the input feature. An improved performance is achived using this classifier over the spectral magnitude based features.

2.4.3 Speech synthesis

Group delay is applied to improve the segmentation of speech into syllables in concatenative speech synthesis. Unit selection synthesis approach requires a huge database with basic units. Here, the objective is to select the best sequence of speech units from all possible contexts. Syllable is a speech unit suited well for Indian languages. Speech is analysed at the syllable-level using GD based segmentation algorithm (Nagarajan *et al.*, 2003). This algorithm is modified so as to detect the vowel onset points with minimum errors owing to insertions and deletions. A tool for labelling the utterances for the acoustic modelling in TTS is developed using this minimum error criterion in Deivapalan *et al.* (2008). This labelling tool is used for six languages, namely, Tamil, Hindi, Bengali, Malayalam, and Telugu. In contrast to phoneme based synthesis systems, syllable based TTS are built in (Pradhan *et al.*, 2010) and in (Vinodh *et al.*, 2010).

An algorithm for phoneme segmentation on HMM-based speech synthesis (HTS) framework is proposed in (Shanmugam and Murthy, 2014*b*). The short-time energy (STE) of the speech waveform is considered as the magnitude spectrum of a minimum phase signal and GD processing is employed to get the syllable boundaries. These boundaries are used to correct the syllable boundaries obtained via a hidden Markov model. This tandem process provides accurate syllable boundaries. The GD based boundaries obtained close to the HMM estimated boundaries are used as the actual boundaries to re-estimate the monophone HMM model. The re-estimated boundaries are again compared with the group delay boundaries and are corrected in an iterative fashion. This tandem approach yielded better qualitative performance compared to standalone HMM system.

2.4.4 Speech emotion recognition

Speech emotion detection systems are being employed in numerous applications such as voice enabled chatbots, humanoid robots, mobile communication, and call centre applications. Group delay function is applied for emotion recognition in (Sethu *et al.*, 2007). The phase spectrum of vocal tract model is used as an additional feature along with pitch, energy and other traditional prosodic features. The all-pole system model is obtained from linear prediction analysis for modelling the vocal tract information. Depression detection from speech is analysed using GD in (Ming *et al.*, 2013). The

benefits of delta phase (MFDP) in comparison to MFCCs is illustrated in this work. In (Dey *et al.*, 2011), the relevant features for emotional classes are automatically selected using feature switching.

2.4.5 Voice activity detection and vocal tract estimation

The presence or absence of human speech in an audio signal is estimated by a Voice activity detector (VAD). It is an essential pre-processing step various speech applications. Speech activity detection using group delay functions is presented in (Hari Krishnan P *et al.*, 2006). Speech regions are manifested as peaks in the GD domain, whereas it shows clear valleys for non-speech regions. Two approaches based on modified group delay is discussed in (Padmanabahan, 2012) for identifying the voice activity. MOD-GDF is used as a feature in both the approaches; a Gaussian mixture model (GMM) based VAD and a multi layer perceptron (MLP) based VAD, and provided reduction in error measures compared to standard approaches.

Chirp group delay is the negative derivative of the phase spectrum computed from chirp z-transform (Bozkurt *et al.*, 2007). Vocal tract estimation approaches proposed in (Jayesh and Ramalingam, 2014) and (Jayesh and Ramalingam, 2016) use a variant of chirp group delay. These estimation methods are less sensitive to the duration and the starting point of the analysis window. They also do not explicitly require the temporal positions of glottal closures for separating them out.

2.4.6 Music information retrieval

Group delay functions have found many applications in music processing. Indian classical music consists of two categories known as Carnatic and Hindustani music. " Rāga, tāla and lyrics form the three pillars on which Carnatic music rests. A rāga is related to melodic modes whereas the Tāla is the repeating rhythmic cycle, which is a measure of time.

Tonic is defined as the base frequency which a performer uses as the reference. All the instruments are tuned to this frequency. Group delay based estimation of tonic is employed in (Bellur, 2013) using pitch histograms via three different methods. In template matching based method, every peak in the histogram is fitted to a template,

which is a candidate tonic. In segmented histogram method, the tonic pitch is estimated by taking the bin-wise product of segmented GD histogram and then finding the tallest peak. Concert based method estimates the tonic by computing the GD histograms for all the items separately and then by multiplying them bin-wise to get a single histogram representation for the concert. The tallest peak on this histogram corresponds to the tonic.

GD based methods are proposed for melody mono-pitch and multi-pitch extraction in (Rajan, 2017). Two methods are presented for estimating the pitch. In the first approach, modified group delay of the spectrum is utilised and it is known as MODGD-Direct. In the second algorithm, MODGD-Source is estimated from the flattened spectrum. The algorithms show great potential as a a stand-alone algorithm for pitch estimation from music signals. A variant of this algorithm is proposed for multi-pitch estimation task. It consists of a two pass system; In the first pass, MODGD-Source is used to estimate the pre-dominant melody pitch. The frequency positions and its harmonics are annihilated from the flattened spectrum using a comb filter centred at pitch and its harmonics. The pitch is then estimated on the resultant signal in the second pass, following the same procedure as in the first pass. The tracking of individual pitches are done using a dynamic programming method. This algorithm shows improvement over the traditional multi-pitch extraction techniques. Methods for classifying the signal into speech/music is also developed in Rajan (2017) using MODGD-Source based features.

2.5 Summary

Group delay functions and the need for minimum and modified group delay functions are discussed in this chapter. The properties of minimum phase group delay functions is first discussed. The standard group delay function suffers from spikes when the roots of the transfer function are close to the unit circle in the z-domain. Two variants of group delay functions, namely minimum phase, and modified group delay functions are reviewed. A review of different applications using GD functions for processing speech signals is presented.

CHAPTER 3

Theoretical Analysis of High Resolution Property of Group Delay Functions

3.1 Introduction

This chapter provides a new insight into the high resolution (HR) property of the negative derivative of the phase response of a system. As discussed in Chapter 2, group delay functions have been proposed and applied successfully as an alternative to conventional magnitude spectrum based applications in speech and music signal processing. One of the reasons claimed for its superior performance is the high spectral resolution. Most of the existing works use empirical analysis to show this property. In this chapter, a mathematical proof of HR property is shown for single and multi-pole resonators. It is established that the ratio of the value of the peak in the magnitude spectrum to the value at a frequency that is n dB below the peak is always much lower than that of the minimum phase group delay spectrum. Numerical analyses are also used for reinforcing the high resolution property. The proof is then extended for multi-pole minimum phase systems.

This chapter is organised as follows. Section 3.2 provides an introduction to the high resolution property. Section 3.3 discusses the previous analyses of the HR property. Section 3.4 considers the case of a single resonator system and shows that the group delay function always possesses a sharper peak in comparison to the magnitude spectrum, without any constraints on the location of poles. Section 3.5 extends the proof for multi-pole minimum phase systems which are connected either in series or in parallel. Section 3.6 quantifies the HR property using numerical computations and empirical measures such as kurtosis and spectral flatness. Conclusions are presented in Section 3.7.

3.2 High Resolution Property

Bandwidth of group delay function is always lesser than that of the magnitude spectrum for minimum phase signals. This property is termed as high resolution property. The sharper peaks in GD domain enable group delay to perform better than magnitude based features in various segmentation tasks in speech signal processing literature.

High resolution property is illustrated with an example of a two-pole system in Figure 3.1. Figure 3.1 (*top panel*) shows the z-plane plot consisting of the locations of two complex conjugate pole pairs. The corresponding magnitude and group delay spectra are shown in Figure 3.1 (*middle panel*) and 3.1 (*bottom panel*), respectively. The pole locations are marked in red colour in the spectra. Observe that the individual poles are resolved better in GD domain compared to that of the magnitude spectrum domain.

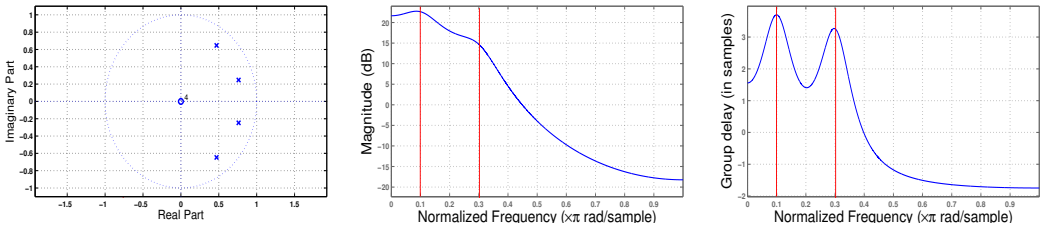


Figure 3.1: Illustration of high resolution property for multi-pole systems. (*top panel*) z-plane representation of a two-pole system, (*middle panel*) magnitude spectrum, and (*bottom panel*) group delay spectrum of the system shown in (*top panel*).

An interpretation for high resolution of multi-pole systems is presented in (Rajan, 2017). Considering a causal and discrete-time signal consisting of two complex conjugate poles, the $z-$ domain representation is given by:

$$X(z) = \frac{1}{(z - z_0^*)(z - z_0)(z - z_1^*)(z - z_1)} \tag{3.1}$$

where * is the complex conjugation operation. The frequency response is given by,

$$X(e^{j\omega}) = \prod_{i=0}^{1} \frac{1}{(e^{j\omega} - e^{-(\sigma_i - j\omega_i)})} \prod_{i=0}^{1} \frac{1}{(e^{j\omega} - e^{-(\sigma_i + j\omega_i)})} \tag{3.2}$$

where, $e^{-\sigma_0}$ and $e^{-\sigma_1}$ determines proximity of the root to the unit circle. The corresponding phase spectrum is given by,

$$\begin{aligned}\theta(\omega) &= -\sum_{i=0}^{1}\tan^{-1}\frac{\sin\omega - e^{-\sigma_i}\sin\omega_i}{\cos\omega - e^{-\sigma_i}\cos\omega_i} + \\ &\quad -\sum_{i=0}^{1}\tan^{-1}\frac{\sin\omega + e^{-\sigma_i}\sin\omega_i}{\cos\omega - e^{-\sigma_i}\cos\omega_i}\end{aligned} \tag{3.3}$$

Differentiating Equation 3.3 with respect ω, it can be shown that:

$$\begin{aligned}\theta^{'}(\omega) &= \theta_1^{'}(\omega) + \theta_2^{'}(\omega) + \theta_3^{'}(\omega) + \theta_4^{'}(\omega) \\ \tau(\omega) &= \tau_1(\omega) + \tau_2(\omega) + \tau_3(\omega) + \tau_4(\omega)\end{aligned} \tag{3.4}$$

where $\theta_i^{'}$ is the derivative of individual terms in Equation 3.3, and $\tau_i(\omega)$ is the group delay functions of each of the single poles, respectively. For simplicity, consider the group delay function of the first two terms alone. Assuming $\sigma_1 = 3\sigma_0$, the group delay for positive half of the spectrum can be obtained as,

$$\begin{aligned}\tau_+(\omega) &= -\frac{1 - e^{-\sigma_0}\cos(\omega - \omega_0)}{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega - \omega_0)} + \\ &\quad -\frac{1 - e^{-3\sigma_0}\cos(\omega - \omega_1)}{1 + e^{-6\sigma_0} - 2e^{-3\sigma_0}\cos(\omega - \omega_1)}\end{aligned} \tag{3.5}$$

To identify the zero crossing points, the derivative of $\tau_+(\omega)$ is taken and is set it to zero.

$$\begin{aligned}\tau_+^{'}(\omega) &= \frac{(e^{-\sigma_0} - e^{-3\sigma_0})\sin(\omega - \omega_0)}{(1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega - \omega_0))^2} + \\ &\quad \frac{(e^{-3\sigma_0} - e^{-9\sigma_0})\sin(\omega - \omega_1)}{(1 + e^{-9\sigma_0} - 2e^{-3\sigma_0}\cos(\omega - \omega_1))^2} = 0\end{aligned} \tag{3.6}$$

1. When $\sigma_0 = 0$, Equation 3.6 becomes zero. This refers to a system in which poles or zeros lie exactly on the unit circle.
2. When $\omega = \omega_0$, the first term becomes zero, and the second term takes values which are much less than one because of $(e^{-3\sigma_0} - e^{-9\sigma_0})$. Group delay function decays fast around ω_0 owing to the presence of $\cos$ and $(e^{-3\sigma_0} - e^{-9\sigma_0})$
3. Finally, when $\omega = \omega_1$, it behaves similar to the case 2, provided $\sigma_0 > 0$ and $\omega - \omega_0, \omega - \omega_1$ are less than π.

3.3 Background

The group delay function was studied in the vicinity of resonance locations, in one of the earlier efforts. In (Yegnanarayana, 1979), formant estimation was attempted from the linear prediction phase spectra, and a cascade of resonators was considered for the interpretation. It was shown that for a constrained location of the poles, the squared magnitude behaviour of the group delay function around the resonance leads to its high resolution property. The magnitude spectrum for a cascade of N poles ($\alpha_i \pm \beta_i$ where $1 \leq i \leq N$) is given by:

$$|H(\omega)|^2 = \prod_{i=1}^{N} \frac{1}{(\alpha_i^2 + \beta_i^2 - \omega^2)^2 + 4\omega^2\alpha_i^2)} \tag{3.7}$$

The corresponding group delay spectrum is:

$$\theta'(\omega) = \sum_{i=1}^{N} \frac{2\alpha_i(\alpha_i^2 + \beta_i^2 - \omega^2)}{(\alpha_i^2 + \beta_i^2 - \omega^2)^2 + 4\omega^2\alpha_i^2)} \tag{3.8}$$

For $\beta_i^2 >> \alpha_i^2$, the group delay can be approximated as

$$\theta'(\omega) \simeq \sum_{i=1}^{N} \frac{K_i}{(\alpha_i^2 + \beta_i^2 - \omega^2)^2 + 4\omega^2\alpha_i^2)}$$

or

$$\theta'(\omega) \simeq \sum_{i=1}^{N} K_i |H(\omega)|^2 \tag{3.9}$$

where K_i is a constant. Hence, most of the energy in the group delay domain is argued to be concentrated around the resonator, thus enabling better formant estimation. This explanation of group delay as a squared magnitude response places a significant constraint on the pole that it must be close to the imaginary axis and have a small bandwidth.

Later methods (Bellur and Murthy, 2013*b*) examined a parallel connection of resonators while working on pitch histograms that resembled a non-constant Q factor for each of the peaks. By studying an example of two resonators in parallel, it was shown that the group delay response was again approximated by the squared magnitude spectrum around the peaks. Both (Yegnanarayana, 1979) and (Bellur and Murthy, 2013*b*) consider only the region around the peaks, and present analyses which consider the

comparisons rather than analysis proof of peakedness. These efforts lie as a motivation for us to mathematically prove the high resolution property.

3.4 Minimum-phase Single-resonator System

This section provides a mathematical proof governing the high resolution property for single pole and minimum phase systems[1]. Consider a causal, discrete-time signal $x[n]$ with one pole whose location in the z-plane is given as $z_0 = re^{j\omega_0}$, or $z_0 = e^{-\sigma_0+j\omega_0}$. The term σ_0 represents the bandwidth of the pole and ω_0 represents the angle with respect to the abscissa. The Z-transform of this system is given by:

$$X(z) = \frac{1}{(z - z_0)(z - z_0^*)} \tag{3.10}$$

The complex Fourier transform is obtained when evaluated at the unit circle:

$$X(\omega) = \frac{1}{(e^{j\omega} - e^{-\sigma_0+j\omega_0})(e^{j\omega} - e^{-\sigma_0-j\omega_0})} \tag{3.11}$$

The expression for the magnitude spectrum is given as:

$$|X(\omega)| = P \times Q \tag{3.12}$$

where

$$P = \frac{1}{\sqrt{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega - \omega_0)}} \tag{3.13}$$

$$Q = \frac{1}{\sqrt{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega + \omega_0)}} \tag{3.14}$$

Considering only (3.13), the maximum value of $\frac{1}{1-e^{-\sigma_0}}$ occurs at an angular frequency of $\omega = \omega_0$. To compute the n dB bandwidth, the ω_1 at which the magnitude spectrum falls to $\frac{1}{N}$ of its maximum value is determined, i.e

$$\frac{1}{\sqrt{(1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega_1 - \omega_0))}} = \frac{1}{N(1 - e^{-\sigma_0})} \tag{3.15}$$

Here, $N = 10^{\frac{n}{20}}$. Solving for ω_1,

[1]Collaborative work with Manoj Kumar P. A. (Kumar, 2015)

$$\omega_1 = \omega_0 \pm \cos^{-1}(N^2 + \frac{1-N^2}{2}(e^{\sigma_0} + e^{-\sigma_0})) \tag{3.16}$$

The n dB bandwidth is the interval with ω_0 at the centre, and is given by

$$\omega_{ndB} = 2\cos^{-1}(N^2 + \frac{1-N^2}{2}(e^{\sigma_0} + e^{-\sigma_0})) \tag{3.17}$$

This analysis is repeated for the group delay spectrum. The phase spectrum for the system defined by (3.11) is given by

$$\theta(\omega) = -\tan^{-1}\left(\frac{\sin(\omega) - e^{-\sigma_0}\sin(\omega_0)}{\cos(\omega) - e^{-\sigma_0}\cos(\omega_0)}\right) - \tan^{-1}\left(\frac{\sin(\omega) + e^{-\sigma_0}\sin(\omega_0)}{\cos(\omega) - e^{-\sigma_0}\cos(\omega_0)}\right) \tag{3.18}$$

The group delay is defined as the negative derivative of the unwrapped phase spectrum with respect to frequency and is given by

$$GD(\omega) = \frac{1 - e^{-\sigma_0}\cos(\omega - \omega_0)}{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega - \omega_0)} + \frac{1 - e^{-\sigma_0}\cos(\omega + \omega_0)}{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}\cos(\omega + \omega_0)} \tag{3.19}$$

Differentiating the first term in Equation 3.19 and equating to zero, it can be observed that it displays the same abscissa and ordinate for the maxima as the magnitude spectrum. Solving for the n dB frequency,

$$\frac{1 - e^{-\sigma_0}\cos(\omega_1 - \omega_0)}{1 + e^{-2\sigma_0} - 2e^{-\sigma_0}cos(\omega_1 - \omega_0)} = \frac{1}{N(1 - e^{-\sigma_0})} \tag{3.20}$$

$$\omega_1 = \omega_0 \pm \cos^{-1}\left(\frac{(1-N) + Ne^{-\sigma_0} + e^{-2\sigma_0}}{Ne^{-2\sigma_0} + e^{-\sigma_0}(2-N)}\right) \tag{3.21}$$

Hence, the n dB bandwidth is given as

$$\omega_{ndB} = 2\cos^{-1}\left(\frac{(1-N) + Ne^{-\sigma_0} + e^{-2\sigma_0}}{Ne^{-2\sigma_0} + e^{-\sigma_0}(2-N)}\right) \tag{3.22}$$

Since n dB bandwidth need not exist for all possible pole locations (i.e, if the half power amplitude is lesser than strength at ω = 0 and $\omega = \pi$), its existence for the case of group delay and magnitude spectrum is discussed separately. The arguments of $\cos^{-1}$

function in Equation 3.17 and Equation 3.22 are constrained to lie within $[-1, 1]$. This leads to a set of constraints on $e^{-\sigma_0}$ in magnitude and group delay spectra. A range of values of $e^{-\sigma_0}$ satisfied by both group delay and magnitude spectra are finally derived as follows:

Magnitude spectrum:

$$-1 \leq N^2 + \frac{1-N^2}{2}(e^{\sigma_0} + e^{-\sigma_0}) \leq 1 \tag{3.23}$$

e^{σ_0} being positive, the expression is always less than 1.

$$N^2 + \frac{1-N^2}{2}(e^{\sigma_0} + e^{-\sigma_0}) \geq -1 \tag{3.24}$$

Solving the quadratic equation in $e^{-\sigma_0}$,

$$e^{-\sigma_0} \in [\frac{N-1}{N+1}, \frac{N+1}{N-1}] \tag{3.25}$$

σ_0 being positive, the effective range of $e^{-\sigma_0}$ is reduced to $[\frac{N-1}{N+1}, 1]$

Group delay spectrum:

$$-1 \leq \left(\frac{(1-N) + Ne^{-\sigma_0} + e^{-2\sigma_0}}{Ne^{-2\sigma_0} + e^{-\sigma_0}(2-N)} \right) \leq 1 \tag{3.26}$$

The inequality gives rise to two quadratic equations:

$$(N+1)e^{-2\sigma_0} + 2e^{-\sigma_0} + (1-N) \geq 0 \tag{3.27}$$

$$(1-N)e^{-2\sigma_0} + (2N-2)e^{-\sigma_0} + (1-N) \leq 0 \tag{3.28}$$

The common range of $e^{-\sigma_0}$ in both of them being:

$$e^{-\sigma_0} \in [\frac{N-1}{N+1}, \infty] \tag{3.29}$$

Equations (3.25) and (3.29) result in $\sigma_0 \in [\frac{N-1}{N+1}, 1]$ as the interval for consideration of n dB bandwidth.

Now, the value of the group delay function at the n dB bandwidth of the magnitude spectrum is considered. Substituting for Equation 3.16 in the first term of Equation 3.19,

$$\tau(\omega) = \left(\frac{(1+N^2) + e^{-2\sigma_0}(N^2-1) - 2N^2 e^{-\sigma}}{2[N^2(1+e^{-2\sigma_0}) - 2N^2 e^{-\sigma_0}]} \right) \tag{3.30}$$

The magnitude spectrum at the same frequency was shown to have a value of $\frac{1}{N(1-e^{-\sigma_0})}$ in Equation 3.17. The difference between this value and that of the group delay spectrum in Equation 3.30 is given by:

$$\frac{1}{N(1-e^{-\sigma_0})} - \frac{(1+N^2) + e^{-2\sigma_0}(N^2-1) - 2N^2 e^{-\sigma}}{2[N^2(1+e^{-2\sigma_0}) - 2N^2 e^{-\sigma_0}]} \tag{3.31}$$

The denominator function is positive. The numerator is a quadratic expression which is positive in the interval $e^{-\sigma_0} \in [\frac{N-1}{N+1}, 1]$, which is the same for the existence of n dB bandwidth. This can be verified as follows:

The numerator and denominator are obtained by taking the L.C.M of the two terms of (3.31) as,

$$\begin{aligned} &\frac{2[N^2(1+e^{-2\sigma_0}) - 2N^2 e^{-\sigma_0}]}{N(1-e^{-\sigma_0})2[N^2(1+e^{-2\sigma_0}) - 2N^2 e^{-\sigma_0}]} \\ &- \frac{N(1-e^{-\sigma_0})[(1+N^2) + e^{-2\sigma_0}(N^2-1) - 2N^2 e^{-\sigma_0}]}{N(1-e^{-\sigma_0})2[N^2(1+e^{-2\sigma_0}) - 2N^2 e^{-\sigma_0}]} \end{aligned} \tag{3.32}$$

The denominator of Equation (3.32) is positive. The numerator can be expressed as,

$$e^{-\sigma_0}[Ne^{-2\sigma_0} + (N^3+N+2N^2)e^{-\sigma_0} + (3N^3-4N^2+N)] - N(N^2-2N+1) \tag{3.33}$$

The expression inside [.] is obtained as a quadratic expression in $e^{-\sigma_0}$, whose value is positive for the given range of $e^{-\sigma_0}$. This is explained below:

Factoring out N from the previous equation,

$$e^{-\sigma_0}[e^{-2\sigma_0} + (N^2+1+2N)e^{-\sigma_0} + (3N^2-4N+1)] - (N-1)^2 \tag{3.34}$$

$$\frac{e^{-2\sigma_0} + (N+1)^2 e^{-\sigma_0} + (3N^2-4N+1)}{(N-1^2)} \tag{3.35}$$

Showing Equation (3.34) as positive is same as showing (3.35) greater than 1. Substituting the minimum value of $e^{-\sigma_0}$ in the above Equation,

$$\geq \frac{\frac{(N-1)^2}{(N+1)^2} + (N+1)(N-1) + (3N^2-4N+1)}{N^2-1} \tag{3.36}$$

Equation (3.36) is less than or equal to Equation (3.35) since $(a-1)^2 \leq a^2 - 1$, and $\frac{N-1}{N+1}$ is the minimum possible value of $e^{-\sigma_0}$.

$$1 + \frac{N-1}{(N+1)^3} + \frac{3N^2 - 4N + 1}{N^2 - 1} \tag{3.37}$$

$$1 + \frac{N-1}{(N+1)^3} + \frac{(3N-1)}{N+1} \tag{3.38}$$

$$1 + \frac{3N^3 + 5N^2 + 2N - 2}{(N+1)^3} \tag{3.39}$$

$$1 + \frac{3N^3 + 5N^2 + 3N + 1}{N^3 + 3N^2 + 3N + 1} - \frac{1}{(N+1)^2} - \frac{2}{(N+1)^3} \tag{3.40}$$

For the second term in Equation (3.40), every term in the denominator is less than or equal to the corresponding term in the numerator. Hence, the value of this term is >1. Third and fourth terms are with values less than one and are subtracted from the equation. For the minimum N value (N=1), they achieve a maximum value of 0.5. The value of Equation (3.40) is strictly greater than 1.5 (1+1-0.5). Hence, the numerator of Equation (3.33) is positive. i.e, the n dB value of the magnitude spectrum is greater than the GD value (calculated at n dB bandwidth of the magnitude spectrum). Hence, n dB bandwidth of GD is always smaller than the magnitude spectrum. i.e., for the same peak amplitudes, the ratio of the value of the group delay function at the location of the resonator to the value at n dB frequency is higher compared to the same ratio of the magnitude spectrum. Figure 3.2 shows the phenomenon explained in this Section for a particular σ_0 in Equation 3.31. When substituted $\omega = \omega_0$ in Equation 3.13 and for the first term in Equation 3.19, the same maximum ordinates are obtained. The group delay values are converted to dB only for comparison purpose.

Note the following points before validating the above statement in the following Section:

- The same analysis can be extended to a single-zero system, in which case the magnitude spectrum is inverted, and the group delay function is negated in sign. Identical expressions for n dB bandwidths (Equation 3.17) and (Equation 3.22) can be derived. The 3 dB example is shown for a single zero system in Figure 3.2 (*right*).
- The above analysis has been carried out on the positive half of the spectrum. The effect due to the conjugate term in Equation 3.11 has not been considered,

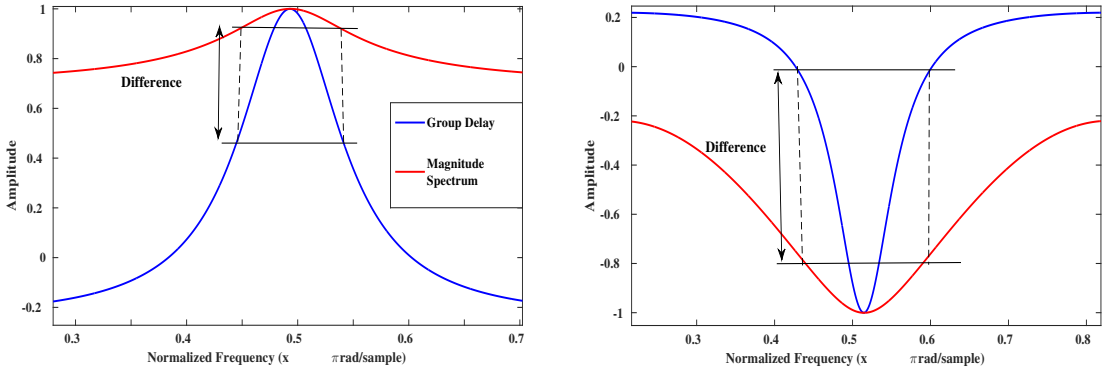


Figure 3.2: Illustrating the high resolution property using 3dB example. A single pole (*left*) and single zero (*right*) systems are shown. Faster decay of group delay in comparison to magnitude spectrum is shown by the difference at the half power frequency of the magnitude spectrum.

although the additive property (Equation 3.19) also contributes to the higher resolution.

- The analysis is based on the minimum phase assumption about the signal. The purpose of this analysis is to compare the width of intervals below the maximum value (or above the minimum for zeros) for the group delay and the magnitude spectrum.

3.5 Analysis for Multi-resonator Systems

In the following Subsections, a theoretical analysis of group delay is presented for multi-pole systems to show that the property of GD functions is preserved for multiple peaks as well. This is the first generalised proof of the high resolution property for minimum-phase, multi-pole systems. Two different combinations of single pole systems that result in multi-pole systems are considered: a) A cascade connection of resonators and b) A parallel connection of resonators.

3.5.1 Cascade connection of resonators

Consider an all-pole system defined by the transfer function in the Z domain as:

$$H(z) = \frac{1}{1 + \sum_{i=1}^{k/2} a_i z^{-i}} \tag{3.41}$$

It is assumed that $H(z)$ can be decomposed into a set of rational polynomials of the

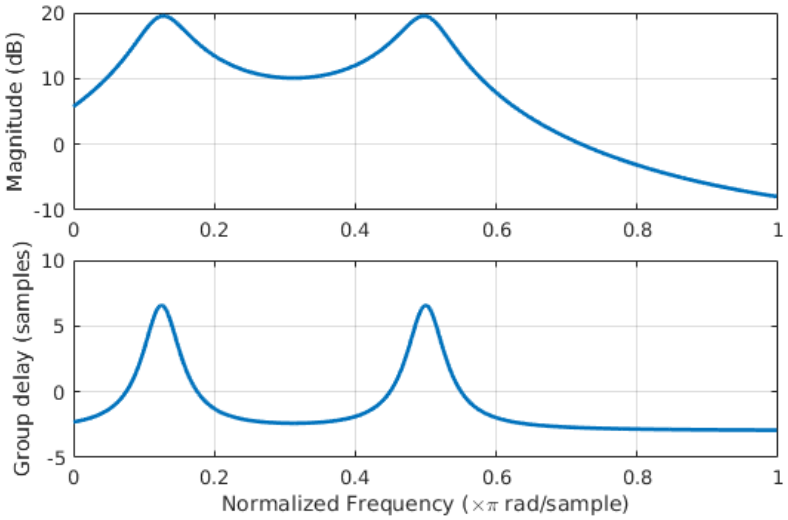


Figure 3.3: Resolving power of the GD function for cascade connection of resonators. *(Top)* Magnitude spectrum and *(Bottom)* group delay spectrum representation.

form $P(z)/Q(z)$. The signal is assumed to be real, thus $P(z)$ and $Q(z)$ can be further factored into complex conjugate order 2 polynomials.

$$H(z) = G.\frac{(z-z_1)(z-z_1^*)...(z-z_{k/2})(z-z_{k/2}^*)}{(z-P_1)(z-P_1^*)...(z-P_{k/2})(z-P_{k/2}^*)} \tag{3.42}$$

where, G is the gain constant and $z_i, i = 1...k/2$ correspond to indices of the zeroes, and $P_i, i = 1, 2..k/2$ are the pole indices.

An all-pole system with an even number of poles is represented by,

$$G.\frac{z^k}{(z-z_0)(z-z_0^*)(z-z_1)(z-z_1^*)...(z-z_{k/2})(z-z_{k/2}^*)} \tag{3.43}$$

while a system with an odd number of poles is given by,

$$G.\frac{z^k}{(z-z_0)(z-z_0^*)...(z-z_{k-1/2})(z-z_{k-1/2}^*)(z-z_k)} \tag{3.44}$$

where $(*)$ is the complex conjugate representation and z_k is the only real pole of the system.

Considering a unity gain constant and assuming that the poles occur in conjugate pairs (corresponding to that of resonances that are not on the real axis),

$$H(z) = \frac{1}{(1 - z_0 z^{-1})(1 - z_0^* z^{-1})...(1 - z_{K/2} z^{-1})(1 - z_{K/2}^* z^{-1})}. \tag{3.45}$$

Assuming that the system is obtained as a product of rational one-pole/two-pole systems, by partial fractions, the frequency response can be represented as a summation of individual responses of each complex conjugate single-pole system,

$$H(z) = \frac{Az^{-1} + B}{(1 - z_0 z^{-1})(1 - z_0^* z^{-1})} + \frac{Cz^{-1} + D}{(1 - z_1 z^{-1})(1 - z_1^* z^{-1})} + ... \tag{3.46}$$

where, A, B, C and D are constant coefficients.

Each term in Equation 3.46 corresponds to a pair of complex conjugate poles, a pole from each pair belongs to 0 to π or 0 to $-\pi$. For every single-pole system in the given $H(z)$, the magnitude spectrum has a lower resolution than the group delay spectrum (Sebastian *et al.*, 2016). If the system responses do not overlap significantly, the overall group delay function can be considered as the addition of the responses of individual poles owing to the property that bandwidth in the group delay domain is inversely proportional to the height (Yegnanarayana, 1979). Hence, for a cascade connection of resonators, group delay exhibits high resolution.

The poles are added in the group delay domain for a multi-pole system obtained by the convolution of single-pole systems. Hence, this additive nature also contributes to the high resolution property. Figure 3.3 shows an example of the magnitude and group delay spectra for a cascade connection of two resonators. The pole locations are at angular frequency locations $\pi/8$ and $\pi/2$ and have a similar bandwidth factor of 0.9. It can be seen that the peak locations are sharper in GD domain. Considering the group delay representation of the individual pole (first term in Equation 3.19), it has a maximum value of $1 - e^{\sigma_0}$ at $\omega = \omega_0$. This decays at values of ω that is away from ω_0 based on the $cos(\omega - \omega_0)$. The $cos(\omega - \omega_0)$ is positive around the pole location as $e^{-\sigma_0} \in [0.1715, 1]$ and causes the response to die off within a range of $\pi/2$.

3.5.2 Parallel connection of resonators

For a parallel connection of resonators, the Z-transform is the addition of individual Z-transforms. Considering a system obtained by the addition of two single pole systems,

$$H(z) = \frac{\alpha_1}{1 - a_1 z^{-1}} + \frac{\alpha_2}{1 - a_2 z^{-1}} \tag{3.47}$$

where, $\alpha_i, i = 1, 2$ is the gain factor associated with the pole $a_i = e^{-\sigma_i + j\omega_i}$. One of the complex conjugate poles is considered for analysis (similar to the single pole analysis in Section 3.4) since only the positive half of the spectrum is analysed (same applies to the negative half). Computing the LCM (least common multiple), (3.47) can be converted to a cascade of resonators again.

$$H(z) = (\alpha_1 + \alpha_2)\frac{1 - C_1 z^{-1}}{(1 - a_1 z^{-1})(1 - a_2 z^{-1})} \tag{3.48}$$

where the constant C_1 is given by,

$$C_1 = \frac{\alpha_2 a_1 + \alpha_1 a_2}{\alpha_1 + \alpha_2} \tag{3.49}$$

Excluding the constant terms, the GD spectrum of the system shown in (3.48) can be written as,

$$GD(H(z)) = GD(1 - C_1 z^{-1}) + GD\Big(\frac{1}{1 - a_1 z^{-1}}\Big) + GD\Big(\frac{1}{1 - a_2 z^{-1}}\Big) \tag{3.50}$$

This equation shows that the overall GD of the system is the summation of GD of single pole/zero systems. Figure 3.4 shows an example of the magnitude and group delay spectra for a parallel connection of two resonators. Pole locations are at frequency locations $\pi/4, \pi/3$ with bandwidth factors of 0.9 and 0.7 respectively. It can be seen that the peak locations are sharper in GD domain for parallel connection of two poles. This interpretation can be extended for multiple resonators by considering the multi-pole system as being obtained by the addition of two systems, where each of the resonators already has the high resolution in the GD domain (commutative property).

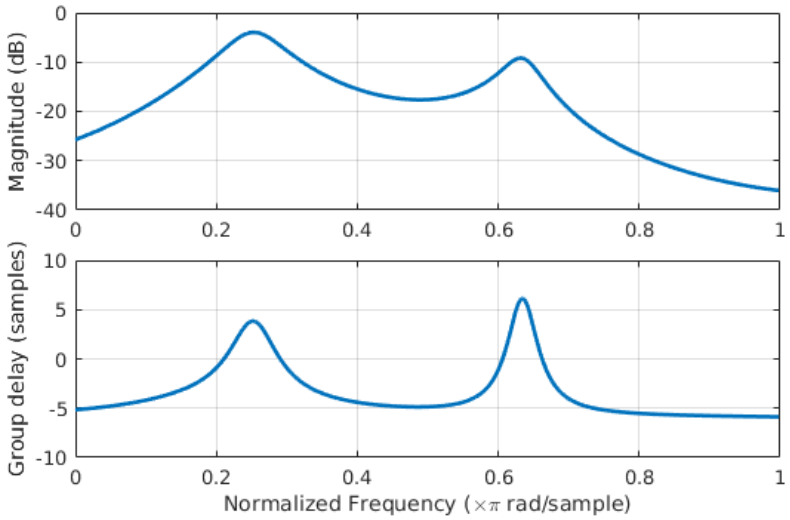


Figure 3.4: Resolving power of the group delay function for parallel connection of resonators. *(Top)* Magnitude spectrum and *(Bottom)* Group delay spectrum representation.

Group delay domain representation for parallel connection of two resonators (Yegnanarayana *et al.*, 1994) is given by,

$$\sum_{i=1}^{2} \frac{2a_i(a_i^2 + b_i^2 - \omega^2)}{(a_i^2 + b_i^2 - \omega^2)^2 + 4\omega^2 a_i^2} \tag{3.51}$$

where the pole is represented by $a_i \pm jb_i$. Substituting the polar form for each of the pole, the numerator becomes

$$2e^{-3\sigma_0} + 4e^{-\sigma_0}\cos(2(\omega - \omega_0)) \tag{3.52}$$

This can also be written as,

$$C_1 + C_2\cos(2(\omega - \omega_0)) \tag{3.53}$$

This equation suggests that the decay from the peak position is proportional to $cos(2x)$ where x corresponds to the distance from the pole. The group delay exhibits high resolution property for various peak heights $(e^{-\sigma_i})$ and decay rates $(cos(\omega - \omega_i))$ with $e^{-\sigma} \in [0.1715, 1]$. This is of great significance for any peak picking task.

3.6 Numerical Analyses

The high resolution property of group delay is demonstrated via numerical analyses on both single pole and multi-pole systems[2]. For single pole systems, kurtosis and spectral flatness measures are considered to verify the sharpness of group delay. Two different window sizes have been used for the experiments. A window size of 512 samples, considering a typical speech frame 25 ms long sampled at 16 kHz/44.1 kHz yields 400/1103 samples, respectively. Another window of 128 samples has also been used to illustrate the property for very short utterances too.

The analyses take the entire frequency spectrum into account. Kurtosis is defined as a scaled version of the fourth standardised moment about the mean of a distribution.

$$k = \frac{E[X - \mu]^4}{\sigma^4} \tag{3.54}$$

Since the spectra is not modelled using a pre-defined distribution, the kurtosis is replaced with a sample kurtosis measure defined as follows:

$$k \approx \frac{\Sigma(X_i - \bar{X})^4)/n}{(\Sigma(X_i - \bar{X})^2/n)^2} \tag{3.55}$$

A distribution with a higher kurtosis value has been argued to represent both higher peakedness and heavy-tailedness ((DeCarlo, 1997)). This property is employed to show the peakedness of the group delay spectrum for a single pole system. For multi-resonator systems, the responses resemble a multi-modal distribution. Hence a single kurtosis value cannot quantify the peakedness of the individual peaks.

Spectral flatness is defined as the ratio of the geometric mean to the arithmetic mean of a power spectrum ((Jayant and Noll, 1984)). It has been used to characterise how noise-like (or tone-like) a waveform is ((Johnston, 1988; Dubnov, 2004)). An Additive White Gaussian Noise (AWGN) signal has a flat spectrum and has the maximum possible spectral flatness value of 1. The more the peakedness, the less the spectral flatness.

The results for kurtosis and spectral flatness obtained by varying the bandwidth (σ_0) and the angular frequency (ω_0) are summarised in Figure 3.5. The behaviour of kurtosis

[2]Collaborative work with Manoj Kumar P. A. (Kumar, 2015)

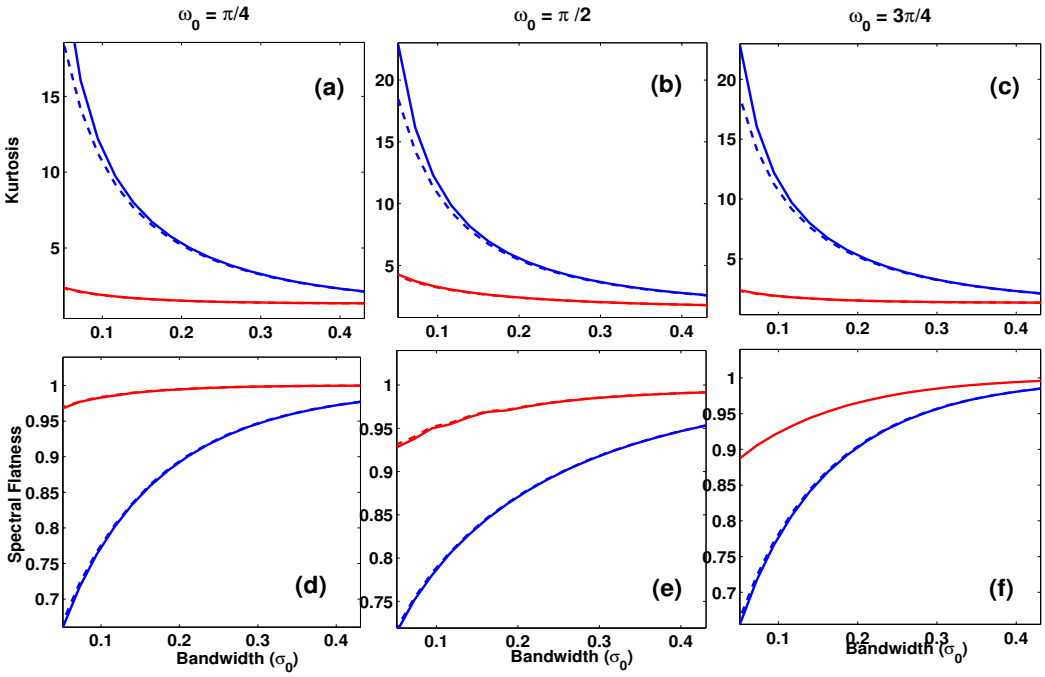


Figure 3.5: Demonstrating the peakedness of group delay functions (*blue*) over log-magnitude spectrum (*red*). Kurtosis measures over a bandwidth range of [0.1,0.4] are shown in (a),(b),(c), respectively. Spectral flatness measures for the same bandwidth range are shown in (d),(e),(f). Dotted lines correspond to the windowed responses.

and spectral flatness measures are noted to be similar irrespective of the pole positions. Both magnitude spectra and group delay converge asymptotically to a spectral flatness of 1 as the bandwidth increases (the pole moving closer to the origin), as expected. The difference in peakedness is emphasised for pole locations with low bandwidths.

In the case of a multi-pole system, the resonant frequency and bandwidth of individual poles define the effect on each other in the spectrum, and hence individual analyses at the poles are more relevant than a global measure. Further, in an n dB analysis, the system can be analysed as multiple single-resonator systems as long as the difference in resonant frequencies is at least twice the n dB bandwidth. Since the group delay value decays exponentially from the pole location, this statement is observed to be true in experiments. In this work, the 3dB bandwidth over all possible configurations ($\sigma \in (0.05, 0.8)$, $\omega \in (0.3\pi, 0.8\pi)$) of the two resonances are computed. Further, the average acceleration measures of group delay and the magnitude spectrum (as a function of frequency) at the vicinity of the poles are used to directly compute the rate of rise and fall around the peaks. The acceleration measure is simply the second order difference function with respect to the DFT bins. It gives an idea of how fast/slow group

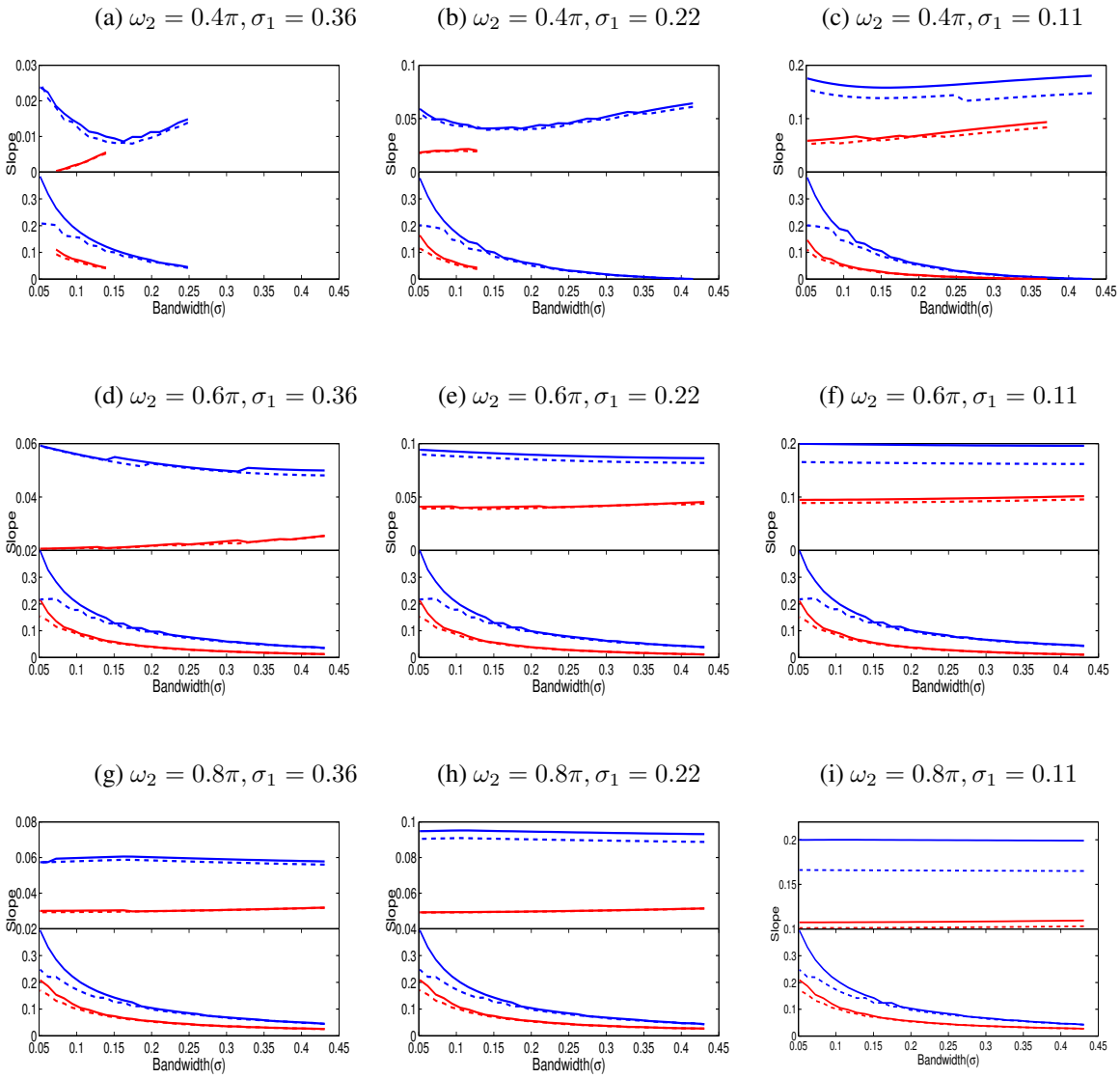


Figure 3.6: Comparison of acceleration measures for group delay (*blue*) and log-magnitude spectrum (*red*). Top half of each figure corresponds to Pole 1 while bottom half corresponds to Pole 2. ω_1 of Pole 1 is kept constant at 0.3π for the entire experiment. σ_2 is varied from 0.44 to 0.05 in each plot. Dotted lines correspond to the windowed responses.

delay/magnitude spectrum changes around the peaks. The pole locations are varied between the same ranges as those in the n dB bandwidth computations. Both numerical 3dB and acceleration (double derivative of the spectra) measures are reported in Table 3.1. The peakedness property is clearly preserved for both measures.

The results are also reported by selecting a few representative configurations for the two poles and varying the bandwidth of the second pole in Figure 3.6. Pole 1 is fixed at an angular frequency of 0.3π (ω_1) in all cases. The bandwidth of Pole 1 (σ_1) and angular frequency of Pole 2 (ω_2) are varied across the various examples, and the bandwidth of Pole 2 (σ_2) is varied within each case. The magnitudes of normalised slopes at each of

Table 3.1: Evaluations illustrating lower amplitude and higher acceleration measure at 3dB frequency of Group delay (GD) over Magnitude Spectrum (MS) for a two pole system.

Measure	GD (in Amplitude)		MS (in Amplitude)	
	pole1	pole2	pole1	pole2
3dB (Hz)	4227.81	3330.31	7430.94	5759.06
Acc.($\times 10^{-2}$)	2.28	2.26	1.13	1.25

the two peaks for log-magnitude and group delay spectra are shown. For configurations where the two peaks are not distinguishable in the magnitude spectrum, the results could not be reported for the entire range of σ_2 (Figure 3.6a, 3.6b, 3.6c). It is observed that windowing operation reduces the acceleration measures for both magnitude spectrum and group delay. This is because the $sinc$ functions of a window increase the bandwidth of both of these functions due to the convolution of the $sinc$ with original spectrum. Yet, group delay continues to possess higher acceleration around the peaks, implying faster decay than the magnitude spectrum.

3.7 Summary

In the current chapter, the importance of group delay functions has been studied with a theoretical perspective with a focus on the greater spectral resolution in comparison with the magnitude spectrum. An analytic study is presented that is corroborated with experimental results. The peakedness of group delay functions has been established and is argued to contribute to the better performance in pitch and formant estimation, spectral reconstruction and speech segmentation tasks.

CHAPTER 4

Time-Event Detection from Speech, Music and Neuronal Signals

Temporal precision in analysis requires sophisticated time-event detectors. Most techniques in the literature that offer high precision use a model based approach to time-event detection (TED). In the earlier chapter, it was shown that the group delay functions obtained from the Fourier transform of a signal have a higher resolution compared to that of the magnitude spectrum obtained from the Fourier transform of the signal. In this chapter, we explore the use of GD functions for a large number of TED tasks, namely, speech, music and spike estimation from neuronal signals. The signal is first pre-processed using group delay based analysis. The signal is further processed to extract useful information, depending upon the domain. During the evaluations, comparison with the state-of-the-art technique is performed wherever possible.

This chapter is organised as follows: Section 4.1 provides an overview of the time-event detection tasks in the literature. The pitch estimation task is then discussed in Section 4.2. The high-resolution property is then employed for percussive onset detection task from music signals in Section 4.3. Section 4.4 discusses the applicability of group delay based time-event detection to spike location estimation from calcium fluorescence traces of neuronal firings. This section further proposes an end-to-end neural network framework for spike estimation. Section 4.5 provides the summary of the tasks discussed. In order to provide the perspective to the proposed approach, the relevant literature is surveyed in the corresponding Sections for each of the tasks.

4.1 Introduction

Temporal resolution is of prime importance in TED tasks. Temporal events invoke the auditory perception of the individuals in the case of an audio signal. For instance, the beginning of every tap on the percussive instrument may signify the start of a stroke, where a sequence of strokes can enable identify patterns, and thereby the rhythm. Like-

wise, instances of significance are defined based on a particular application. They are called time-related events as the events are in various temporal scales or if they change with time. Various segmentation tasks in audio signal processing come under time-event detection.

Methods for time-event detection vary with the application. However, one unique similarity which all the algorithms must possess is that irrespective of the task, the objective is to extract locations of the time events. Three time-event detection tasks are discussed which are at various temporal resolutions. One example each is chosen from speech, music and neuronal signal processing domains. The high-resolution property of group delay functions are discussed in Chapter 3.

We discuss the pitch estimation task from speech signals as a time-event detection task since the relative change with respect to time can be considered as a time-event. The time-scale varies from 4-10 ms in general. In the second task, the beginning of a musical stroke or note is detected from a percussive instrument, exploiting the high resolution property of group delay. The time-scale of onsets varies from milliseconds to seconds.

The ability of GD to enhance the peak locations is agnostic to the signal at hand. However, this is not explored beyond speech and audio domains in the literature. Recording of calcium concentration of a neuron across time using fluorescence imaging creates a slowly varying signal. This signal corresponds to the action potentials (APs) generated by the neuron. Estimating the APs from the observed calcium concentration measure is a challenging problem in neuroscience, and it is referred to as spike estimation task. Challenges such as unknown noise levels, the non-linear relationship of observed fluorescence change with respect to the calcium concentration, baseline fluorescence variation and model assumptions limit the performance of existing algorithms. GD based filtering can provide an improvement over the existing algorithms upon post-processing. It can also be used as a stand-alone algorithm with equivalent performance to the advanced algorithms in the literature. In the following sections, each of these tasks is discussed. Greater focus is given to neuronal spike estimation task owing to the novelty of the domain.

4.2 Pitch Estimation

Pitch estimation is an essential task in several speech applications. For speech signals, the pitch is defined as the fundamental frequency caused by the vibrations of vocal folds. Figure 4.1 shows an example of a speech signal and the estimated pitch locations obtained by the robust algorithm for pitch tracking (RAPT) approach (Talkin, 1995) using Wavesurfer tool (W.S-URL, 2012). This algorithm is contained in a pitch tracker system known as entropic speech processing system (ESPS) pitch tracker.

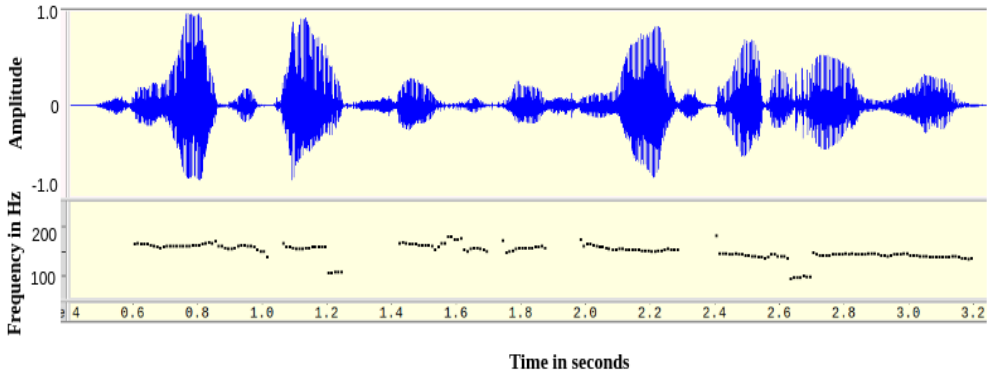


Figure 4.1: (a) A speech utterance, and its corresponding pitch estimates using RAPT algorithm (using Wavesurfer tool).

Pitch estimation is a well-researched subject. It remains to be a challenging task owing to the aberrations from the idealism in the generation of voiced sounds in the glottis area. This Section presents an approach for pitch extraction based on Grating Compression Transform (GCT) on harmonically-enhanced Modified Group Delay based representation.

4.2.1 Introduction to pitch estimation

The first task that we focus on in time-event detection (TED) is the pitch estimation task in which ability of group delay to estimate sharper peaks and pitch dynamics nature of 2D Fourier transforms are used for efficient estimation of pitch locations. Though pitch is a frequency-related quantity, relative change with respect to time is critical and can be considered as a time-event as the vocal folds generate time-varying vibrations. GD based pitch estimation is further enhanced by exploiting the pitch dynamics of a speech signal.

The proposed pitch estimation approach explores the group delay processing of flat-

tened spectrum and the ability of GCT to merge harmonically related elements together in the spectrum. MODGD-source (refer to Section 2.3.2) is extracted from each frame of speech signal. We consider the modified group delay function as a function of time. This is similar to that of the magnitude spectrum as a function of time which is referred to as the spectrogram. This function is therefore referred to as the MODGDgram. Localised time-frequency regions of the MODGDgram are used for computing the GCT. Peak picking is performed on the resulting rate-scale domain to finalise the pitch values.

4.2.2 Overview

Several approaches have been proposed to estimate pitch from speech segments. They can be broadly classified as spectral, temporal or a combination of both. A major challenge in pitch extraction is to suppress noise interference and system information (vocal tract related information) and detect the predominant pitch sequence. A detailed analysis of some of the pitch estimation techniques can be found in (Sondhi, 1968). An approach for pitch estimation based on cepstral representation is presented in (Noll, 1967) whereas a method based on spectral harmonicity is proposed in (Seneff, 1978). RAPT (Talkin, 1995), Yin (Cheveigne and Kawahara, 2002.), and YAAPT (Kasi and Zahorian, 2002) are approaches which use correlation-based methods with a set of post-processing steps to find the pitch values. In (Rao and Rao, 2010), an approach for melody extraction in the presence of pitched accompaniment in polyphonic music is proposed, and an interface for melodic pitch extraction from polyphonic music is discussed in (Rao and Rao, 2010). Salamon et. al. (Salamon and Gómez, 2012) used pitch contour characteristics for pitch estimation from music signals. Latest methods such as (Gonzalez and Brookes, 2014) combine the filtering techniques with temporal continuity constraints to obtain noist-robust pitch estimates.

Phase spectrum has rarely been employed in pitch extraction from speech (Smits and Yegnanarayana, 1995). Nevertheless, modified group delay is found to possess useful properties which make it convenient for this task (Yegnanarayana *et al.*, 1991). The modified power spectrum is processed for pitch estimation from noisy speech using the modified group delay function in (Yegnanarayana *et al.*, 1991) based on the results in (Yegnanarayana and Murthy, 1992). In (Rajan and Murthy, 2013), the concept is extended for pitch extraction from music signals.

In (Quatieri, 2002), application of 2D processing of speech to pitch estimation is proposed. Localised regions of spectrogram are processed by 2D Fourier transform to obtain the GCT of those patches. It was shown to have better performance as compared to sine wave based pitch estimator (McAulay and Quatieri, 1990). Multi-pitch estimation techniques (Wang and Quatieri, 2009) have been proposed on GCT owing to the discriminative ability of GCT in the rate-scale domain. The proposed pitch estimation algorithm is compared with following two advanced methods of pitch estimation and tracking:

Get f_0 algorithm

This algorithm is included in the entropic speech processing system (ESPS) package in Wavesurfer software and this method is an implementation of the robust algorithm for pitch tracking (RAPT) (Talkin, 1995). Wavesurfer uses ESPS pitch tracker, which employs RAPT, an algorithm based on normalized cross-correlation function (NCCF). The approach consists of five steps:

- Provide two versions of sampled speech data; one with original sampling rate; another at reduced sampling rate.
- Periodically compute NCCF of the low sampling rate signal for all lags in the fundamental frequency (f_0) range of interest. Record the locations of local maxima in the first-pass NCCF.
- Compute the NCCF of the high sampling rate signal only in the vicinity of the promising peaks, found in the first pass. Search again for the local maxima.
- Each peak retained in the high resolution NCCF generates a candidate f_0 for that frame.
- Perform dynamic programming across all the frames.

Pitch listing algorithm

This is an implementation of the well established PRAAT algorithm (Boersma, 2001). In PRAAT, pitch extraction can be performed using auto-correlation, cross-correlation, SPINET (Spatial PItch NETwork) or subharmonic summation. The cross-correlation method has better error performance in comparison to other methods in PRAAT.

4.2.3 Performance measures

Performance measures for a pitch estimation algorithm should be able to consider the difference between estimated and original pitch under various conditions. The similarity should exist in frame-level exactness of the values with respect to ground truth. The methods are compared in a pitch estimation task based on the following error measures (Rabiner *et al.*, 1976):

The ***Gross Pitch Error (GPE)*** is the fraction of frames, where the decisions of both the pitch tracker algorithm and the ground truth are voiced, and for which the relative error of f_0 is higher than a threshold of 20%. Errors such as octave error, wrongly estimated pitch due to prominent noise energy at a different frequency etc. comes under this category.

The ***Fine Pitch Error (FPE)*** is defined as the standard deviation (in percent) of the relative error of f_0 for which this error is below a threshold of 20%. These are the regions of the speech waveform during which pitch actually gets tracked. This also measures how closely the algorithm could predict the actual pitch trajectory.

The ***V-UV Error*** is the number of voiced positions which are obtained as unvoiced from the pitch tracker. This accounts for the ability to perform voiced activity detection (VAD) by a pitch estimation algorithm. Owing to this measure, some of the algorithms has explicit VAD estimation algorithms included in the pitch estimation approach.

The ***UV-V Error*** is the number of unvoiced positions which are estimated as voiced by the pitch estimator. This also refers to a voicing error. This could arise due to noise or algorithmic assumptions which results in false positives.

4.2.4 Grating compression transform

Grating compression transform (GCT) is originally introduced in image signal processing domain. It is the localised 2D Fourier transforms of images. 2D Fourier transform of an angled grating is transformed to a dot with a specific angle in the GCT domain. For a time-frequency image such as the spectrogram, the grating-like structure could be observed for a small patch of the image owing to the presence of fundamental frequency and its harmonics. This is shown in Figure 4.2. Observe that the angle of the gratings is well-represented in the GCT domain. Fourier transform of time-frequency domain

results in the rate-scale domain.

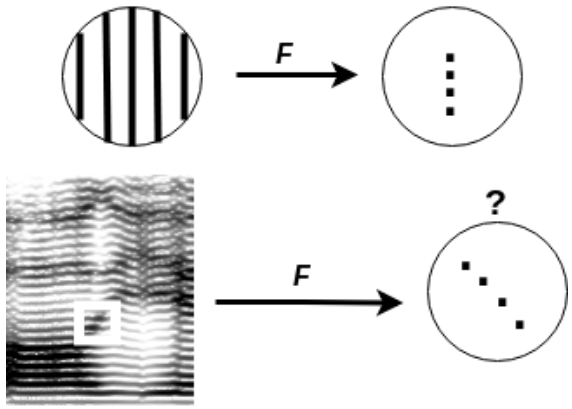


Figure 4.2: Grating compression transform in image signal processing. Gratings are converted to points in the rate-scale domain with an angle proportional to the angle of the gratings.

This analysis is first employed for speech signal processing in (Quatieri, 2002). It is used for estimating the pitch locations from a speech signal. The algorithm considers the spectrogram representation as a 2D sinusoidal function $s[n, m]$ sitting on a flat pedestal.

$$s[n, m] = K + cos(\omega_s \Phi[n, m]) \tag{4.1}$$

where ω_s is the frequency of the sinusoid, n and m are temporal and frequency variables respectively, $\Phi[n, m]$ denotes the spatial orientation and K is an additive constant in the time-frequency domain. The 2D Fourier transform of Equation 4.1 is shown to be:

$$\begin{aligned} S(\omega, \Omega) = 2\pi K\delta(\omega, \Omega) + 2\pi\delta(\omega + \omega_s sin(\theta), \Omega - \omega_s cos(\theta)) \\ +2\pi\delta(\omega - \omega_s sin(\theta), \Omega + \omega_s cos(\theta)) \end{aligned} \tag{4.2}$$

where θ is the orientation of the sinusoid with respect to temporal axis. This means that the 2D Fourier transform of localised regions of spectrogram consists of an impulse at the origin corresponding to flat pedestal and impulses at $\pm\omega_s$ corresponding to sine wave. This is represented in a schematic in Figure 4.3. This example considers increasing pitch where the rotated lines are uniformly spaced over localised time-frequency plane. F denotes 2D Fourier transform.

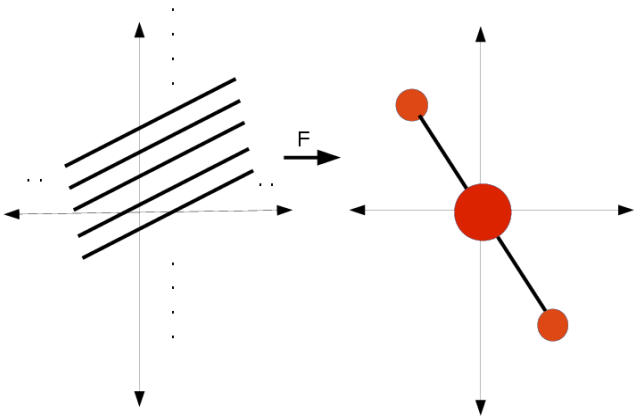


Figure 4.3: Schematic of a harmonic structure and it's 2D Fourier transform.

Since in the GCT domain (referred also to as rate-scale domain) harmonically-related components of the spectrogram are merged together, it is exploited by (Quatieri, 2002) in pitch estimation. Pitch is estimated by peak picking in the rate-scale domain using the equation:

$$f_0 = \frac{1}{N_{DFT}} \frac{2\pi F_s}{\omega_s cos(\theta)} \tag{4.3}$$

where, f_0 is the pitch estimate, N_{DFT} is the order of the DFT used to compute the spectrogram and F_s is the sampling rate of the speech signal.

4.2.5 GD based pitch estimation using GCT

In Subsection 4.2.4, GCT was shown as a powerful transform to locate the pitch along with its direction of trajectory either in positive or negative slope. The method proposed in this work uses a new variant of modified group delay function (after spectral harmonic reinforcement) to build a time-frequency (T-F) representation. It is referred to as the modified group delay-gram (MODGDgram), on which GCT is computed. It differs from earlier applications of GCT which were applied to the magnitude spectrum and also from modified group delay for pitch estimation. The proposed algorithm hence benefits from the ability of GD-based processing to obtain sharper peaks and pitch dynamics of GCT as it performs GCT on the MODGD based representation.

MODGDgram is the short-time modified group delay computed across a set of time frames. The MODGDgram is calculated on the source information for this particular task. System information is excluded by dividing the spectrum by its smoothed version.

The source MODGDgram has a better discriminatory power between the harmonics than the magnitude spectrogram. The pitch estimate can be obtained from modified group delay frames as $1/T_0$, where T_0 is the pitch time-period. The T-F window for computing GCT is centred at the peak location corresponding to the pitch. Since GCT is performed on MODGDgram, it is important to note that the Y-axis in the MODGDgram is also time. Hence, in the rate-scale domain, the vertical distance is *directly* proportional to the value of the pitch in samples.

In Figure 4.3, it is seen that the slope of the impulse from the origin is a measure of the rate of change of pitch for the selected region. This slope measure is obtained as follows.

$$\theta = \tan^{-1}\left(\frac{d_{horizontal}}{d_{vertical}}\right) \tag{4.4}$$

where, $d_{horizontal}$ and $d_{vertical}$ represents horzonatal and verical components of the distance d from the origin. This is used to compute the mean pitch period for each segmented region of MODGDgram. The issue with the pitch estimation algorithm based on GD is that it has errors due to outliers. Most of the errors are committed by a small number of frames for which the deviation is higher than 25 Hz with respect to the ground truth. These errors happen on a per-frame basis. Pitch dynamics property of GCT is used in this algorithm to tackle this problem. As speech is an inertial system, pitch value cannot be very different between two successive frames unless they are voiced-to-unvoiced or unvoiced-to-voiced transitions. Short patches for GCT computation are selected such that, they accommodate small variations in pitch, but with a constant slope (or zero acceleration). Issues caused by outliers is mitigated by this "smoothing" across the frames.

Figure 4.4 demonstrates the pitch estimates obtained from the algorithm for a synthetic signal example. A single frequency sinusoid is the signal used for illustration. Two specific regions of the output and corresponding rate-scale representations are shown in the *bottom* panel. Two areas are magnified for illustrating the GCT representations, which show peak locations as well as slope from the vertical axis, indicating that the GCT captures pitch dynamics.

Source characteristics are seen more clearly in the source MODGDgram when compared to that of the spectrogram in Figure 4.5. A short duration of synthetic speech

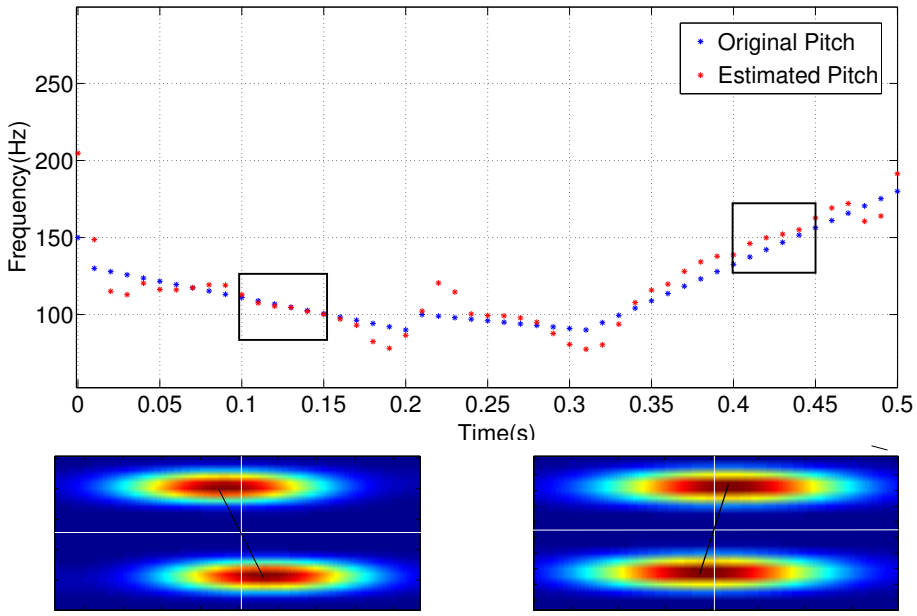


Figure 4.4: Estimated and the original pitch for synthetic signal (*top panel*). Bottom panel shows the GCT representation of MODGDgram computed on a patch of (*left*) decreasing pitch and and (*right*) increasing pitch trajectories.

signal with the pitch trajectory superimposed over them is shown. Observe that the pitch tracks are emphasised at kT_0, where k is an integer for both of the 2D structures. However, the peaks corresponding to pitch are more emphatic in the source MODGDgram compared to that of the spectrogram. Hence, a pitch estimation using source MODGDgram in conjunction with GCT would be better than each of the individual baselines.

4.2.6 Experimentation

Experiments are performed to verify the theoretical findings on the usage of group delay and GCT. Experiments are done on both synthetic and natural speech databases. Synthetic dataset is primarily used as a sanity check of the proposed algorithm, and is not used for evaluation purposes. Natural speech is used for evaluation.

The synthetic dataset has been generated as follows: Assuming a source system model for speech production, a time dependent vocal tract filter was obtained using a formant vocoder model given by:

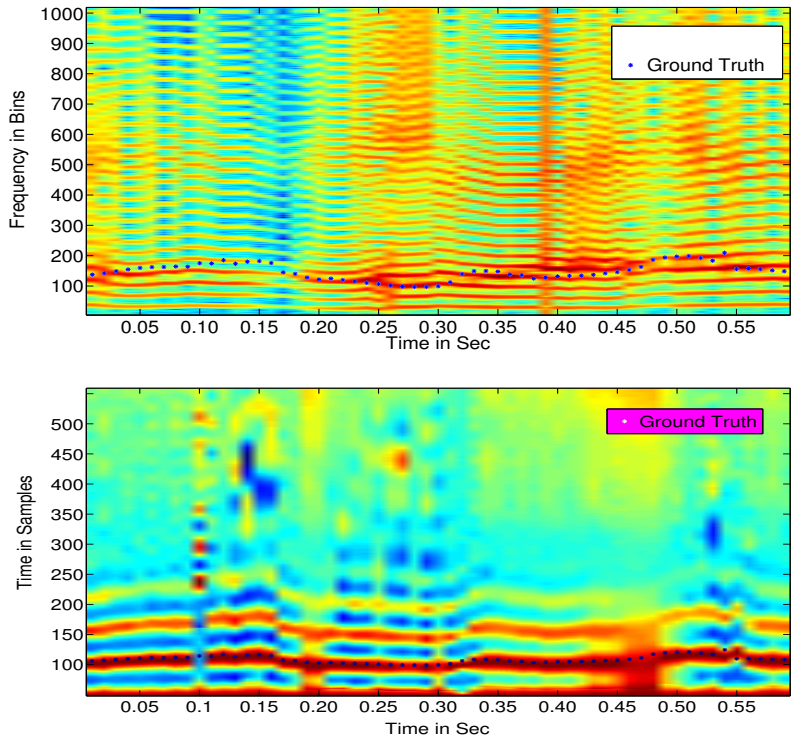


Figure 4.5: Ground Truth plotted on (*top panel*) the spectrogram, and (*bottom panel*) the MODGDgram for synthetic speech.

$$V(z) = \prod_{1}^{3} \frac{1 - 2e^{-\alpha_k T}\cos(2\pi F_k T) + e^{-2\alpha_k T}}{1 - 2e^{-2\alpha_k T}\cos(2\pi F_k T)z^{-1} + e^{-2\alpha_k T}z^{-2}} \tag{4.5}$$

The formant contour used are shown in Figure 4.6. The bandwidths corresponding to α_k were set to 10, 30, 20% of the formant frequency, respectively. This contour was excited by the pitch contour (impulse and Rosenberg's glottal pulse (L. R. Rabiner and R. W. Schafer, 1978)) given in Figure 4.6 (*right*). The frame length was fixed at 256 samples and the sampling rate was set to 16 KHz.

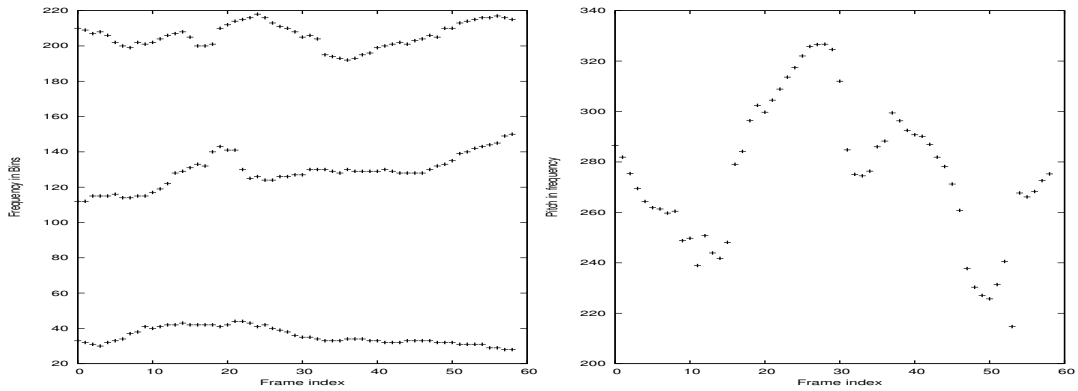


Figure 4.6: Formant (*left*) and pitch (*right*) used to generate synthetic speech.

The natural dataset is Keele pitch dataset (Plante *et al.*, 1995), which is a speech database with corresponding laryngogram recordings marked as the ground truth. This dataset consists of 5 recordings each for men and women speaking the same sentence. The length of each file is a little over 30 seconds. The data are sampled at 20 KHz with a frame length of 25.6 ms and frame shift of 10 ms. The pitch reference provided via laryngogram contains information about voiced and unvoiced frames and it is treated as ground truth. Pitch ranges for the synthetic dataset was fixed at 200-350 Hz and 60-400 Hz for Keele dataset. GCT is computed over segments of 50 ms duration (patches of 5 frames) with 10 ms shift and a frequency range of three times pitch period index. These values are fixed empirically for pitch estimation from speech signals. Pitch period index is a rough estimate of the initial peak and is computed over the MODGD function. The number of DFT points for computing GCT is taken as 4096 along the frequency axis and 512 along the time axis. This ensures a highly resolved source spectrum, which results in the accurate estimation of pitch in the rate-scale domain.

The performance is compared in terms of pitch estimation error measure (Subsection 4.2.3). This includes both the voicing errors and the errors in pitch values. The tracking ability is estimated using FPE and the outlier estimates are reflected in GPE.

Methods compared in this work

The proposed method $(MODGD + GCT)$ is compared with both baseline and advanced algorithms. The baseline algorithms use either GCT or modified group delay. Advanced algorithms are the established pitch estimation algorithms which provide very good pitch estimates. The existing algorithms chosen are methods involving:

$Mag+GCT$**:** This is the implementation of algorithm proposed by (Quatieri, 2002) for single pitch extraction using grating compression transform. This is discussed in Subsection 4.2.4.

$MODGD$**:** This is the algorithm used for pitch extraction from speech (Murthy and Yegnanarayana, 2011) as well as music (Rajan and Murthy, 2013) using modified group delay function.

The proposed method is then compared with two advanced algorithms, Get f_0 and Pitch Listing methods. These methods are discussed in detail in the Subsection 4.2.2.

Results

Table 4.1: Error values for the proposed algorithm versus the baselines.

	Synthetic Database				Keele Database			
Method	**GPE**	**V-UV Error**	**UV-V Error**	**FPE**	**GPE**	**V-UV Error**	**UV-V Error**	**FPE**
$MODGD$	0.038	0.000	0.000	3.857	0.031	0.046	0.213	4.520
$Mag + GCT$	0.051	0.780	0.000	5.240	0.124	0.322	0.113	5.089
ModGD+GCT	0.000	0.076	0.000	3.620	0.015	0.033	0.184	4.165

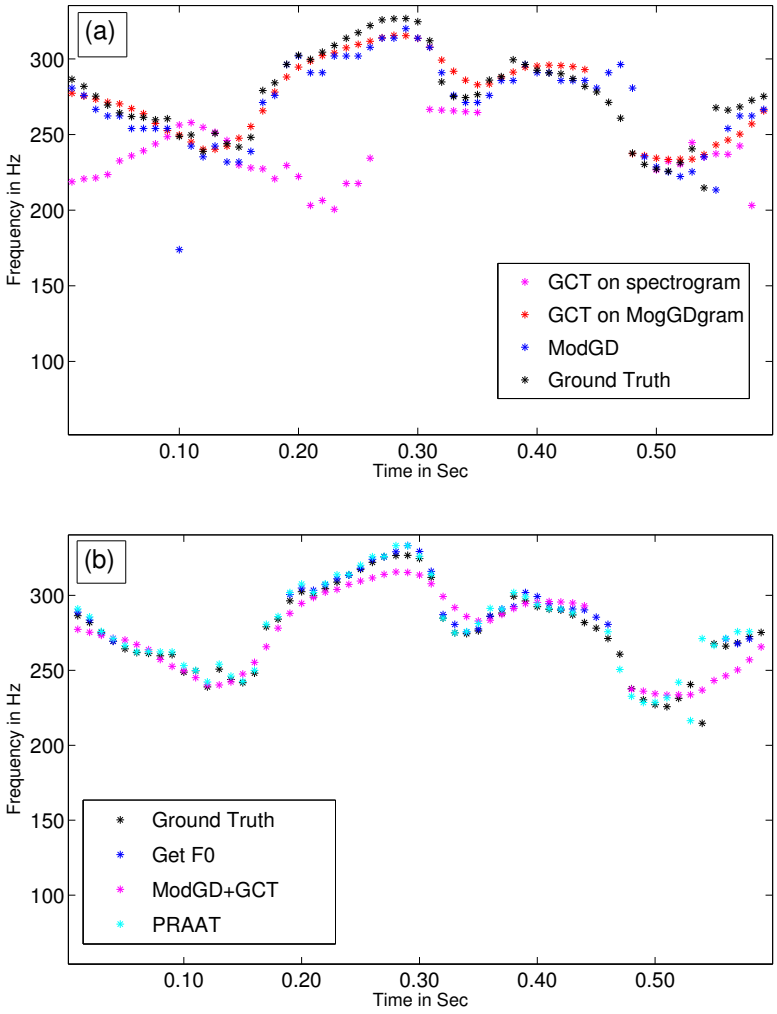


Figure 4.7: Pitch estimated on synthetic signal. Proposed Method is compared with (a) similar baseline approaches and (b) two advanced algorithms ($Get\ f_0$ and $Praat$).

Table 4.1 shows the performance of the proposed algorithm in comparison with similar approaches. The proposed MODGDgram-based algorithm performs much better than the raw-magnitude based approach. This is because the MODGD-source has sharper peaks than that of the magnitude-based source signals and GCT smears these

sharper peaks together, which are harmonically related. Moreover, the MODGDgram is harmonically reinforced, and pitch dynamics are explored for selecting pitch track in GCT domain. Figure 4.7 (a) illustrates the performance of the algorithms on a synthetic dataset. It closely follows the ground truth and has an average improvement of 32% for the synthetic dataset and 21% for Keele dataset over the baseline methods.

Table 4.2: Comparison of the proposed approach with the advanced algorithms ($Get\ f_0$ and $Praat$).

	Synthetic Database				Keele database			
Method	**GPE**	**V-UV Error**	**UV-V Error**	**FPE**	**GPE**	**V-UV Error**	**UV-V Error**	**FPE**
$Get\ f_0$	0.000	0.362	0.000	1.175	0.008	0.029	0.033	2.874
$Praat$	0.017	0.009	0.000	1.901	0.009	0.061	0.050	3.251
ModGD+GCT	0.000	0.076	0.000	3.620	0.015	0.033	0.184	4.165

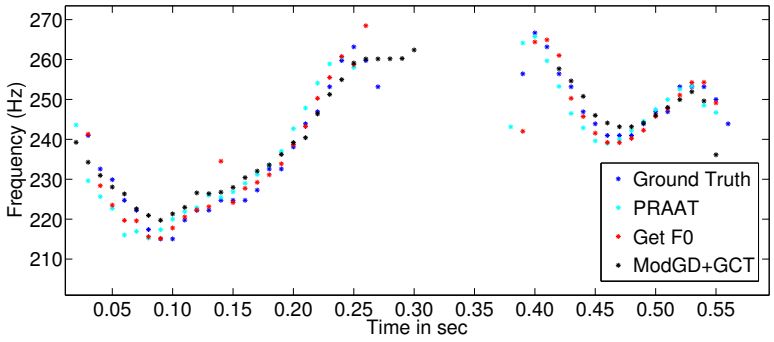


Figure 4.8: Pitch estimated on a segment of female speaker from Keele database. Proposed Method is compared with two advanced algorithms.

The error values for the proposed algorithm are nearly comparable to the advanced algorithms even without a post-processing step (Table 4.2). Figure 4.7 (b) shows the pitch trajectory for a synthetic dataset and Figure 4.8 compares the path for a segment of speech from a female speaker from Keele dataset (natural speech). Even without any voiced activity detector (VAD), the proposed algorithm has similar voiced error values to that of the advanced algorithms. The ability of GD to enhance spectral peaks of the source spectrogram is thus favourably used in conjunction with the pitch dynamics to yield an improved pitch estimation algorithm.

4.2.7 Conclusion to pitch estimation

This section provided an algorithm for pitch estimation from speech signals inspired by the peak-resolving nature of modified group delay and pitch dynamics property of

GCT. Relative change of pitch with respect to time is considered as a time-event and it is then detected by exploiting the GD feature. The precision and relevance of the detected time-event is obtained by comparing the performance with a magnitude spectrum based approach and a purely modified group delay based approach. Further, we show that performance is comparable to existing advanced techniques.

4.3 Percussive Onset Detection

The use of group delay for the musical onset detection is explored for the percussion instruments which are used in Carnatic music. Five major percussion instruments are explored for this task. They are the mridangam, ghatam, kanjira, morsing and thavil. The structure of strokes and their temporal characteristics are discussed, motivating the challenge in designing a percussive onset detection algorithm. A non-model based algorithm is proposed using minimum phase group delay for this task. The music signal is treated as an Amplitude-Frequency modulated (AM-FM) waveform, and its envelope is extracted using the Hilbert transform. Minimum phase GD processing is then applied to determine the onset locations accurately. The algorithm is tested on a large dataset with both controlled and concert recordings (*tani avarthanams*).

4.3.1 Introduction to percussive onset detection

In the context of music signals, Music Information Retrieval (MIR) for estimating the rhythmic characteristics is essential. A study of percussive instruments used in the Carnatic music on the lines of MIR is attempted. The information obtained can be further extended to study higher-level problems such as tāla classification, artist and song recognition and so on. Detecting the beginning of a musical stroke for a percussive instrument is fundamental in estimating other rhythmic characteristic of that percussive instrument. Percussive onsets are considered as a time-event which is essential in estimating higher-level time-events such as sāma detection. To this end, the high-resolution property of GD is explored on a modified envelope of the musical signal to extract onset locations from percussion instruments, with a focus on Carnatic music.

4.3.2 Background

Percussion instruments keep track of rhythm and also produce individual performances which are rich in artistic quality. The major percussive accompaniments to Carnatic (South Indian classical) music include the mridangam, ghatam, kanjira, morsing and the thavil. The thavil, for example, is widely found in musical performances associated with traditional festivals and ceremonies as well as in professional musical concerts. Each of the instruments may be played along with the lead artist (usually a vocalist) or individually (*tani avarthanam*). A study of these instruments on the lines of Music Information Retrieval (MIR) is mostly aimed at first understanding the patterns involved in the performance - beat tracking, stroke classification, phrase/syllable classification, etc. The information obtained can be further extended to study higher-level problems such as tala classification, sama detection, artist and song recognition and so on.

Audio onset detection is characterised as detecting relevant musical events in general, and as identifying the stroke instants in the case of percussive instruments. An onset is defined as the instant chosen to mark the transient (Bello *et al.*, 2005) in the case of a single note, although in most cases it coincides with the start of the transient for impulsive instruments such as percussion instruments. Onset detection can be extended to polyphonic and ensemble performances as well. Various algorithms including knowledge-based post-processing have been proposed, but almost all onset detection algorithms consist of two sub-tasks - extraction of a detection function, and a peak-picking algorithm.

Approaches based on magnitude (Schloss, 1985) and energy (Goto and Muraoka, 1996) report onsets as instants of high absolute or change (Masri, 1996; Böck and Widmer, 2004) in the detection function. Phase-based approaches (Bello and Sandler, 2003) were introduced to detect soft onsets followed by a combination of both energy and phase in (Bello *et al.*, 2004). Multi-resolution analyses with varying window sizes have been tried to study both higher and lower frequency components (Duxbury *et al.*, 2004). Linear prediction error has also been used as a detection function in onset detection (Lee and Kuo, 2006; Gabrielli *et al.*, 2011; Marchi *et al.*, 2014*a*). Recently, the state-of-art techniques employ recurrent (Marchi *et al.*, 2014*b*) and convolutional (Schlüter and Böck, 2014) neural networks for training. Bello (Bello *et al.*, 2005), Dixon (Dixon, 2006) and Böck (Böck *et al.*, 2012) analyse the task in detail and evaluate the major

onset detection approaches over the years.

4.3.3 Convolutional neural network (CNN) for onset detection

In (Schlüter and Böck, 2014), convolutional neural network is introduced for audio onset detection. This method considers onset detection as a binary classification task during the training process. The classifier consists of a CNN trained with 80-band Mel filter banks scaled logarithmically in the magnitude from spectrograms of multiple resolutions. The CNN architecture has convolution and max-pooling in turns compute a set of 20 feature maps, followed by a dense layer with 256 units and sigmoid non-linearity, and an output sigmoid unit. It classifies every frame of input music signal to either onset or non-onset. 102 minutes of monophonic and polyphonic instrumental recordings are used for training the network. The output activation function is smoothened, and a local threshold is used to detect onsets. This approach gives state-of-the-art musical onset detection, with a maximum F-score of 90%.

Percussive onset detection can be performed by training the network with percussive onsets alone. Such a system has an improved F-measure over the standard datasets owing to the absence of very soft onsets. Note that this approach is supervised and has improvement over previously proposed onset detectors for percussive instruments as well.

4.3.4 Performance measures

An onset is treated as correct (*True Positive*) if it is reported within a threshold (± 50ms) of the ground truth. The tolerance is introduced to correct for errors in manual annotation. A *False Positive* does not fall within the threshold of any of the time instants in ground truth wheres a *False Negative* is a missed onset. The following metrics are used to evaluate an onset detection algorithm:

$$Precision(P) = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{4.6}$$

$$Recall(R) = \frac{N_{FN}}{N_{TP} + N_{FP}} \tag{4.7}$$

$$Fmeasure = \frac{2xPR}{P + R} \tag{4.8}$$

N_{TP}, N_{FP} and N_{FN} represent the number of true positives, false positives and false negatives respectively. Onset detection algorithms such as (Eyben *et al.*, 2010; Böck *et al.*, 2012) have merged closely spaced onsets (most commonly, within $30ms$) to match the human perception of onsets, based on psycho-acoustical studies (Handel, 1989). The arithmetic mean was taken and replaced the multiple onsets in such cases.

4.3.5 GD based percussive onset detection

The segmentation ability of group delay is extended to onset detection task in this work[1]. Group delay based onset detection algorithms have been developed in the past (Holzapfel *et al.*, 2010; Böck and Widmer, 2013). However, they do not emphasis the amplitude and frequency characteristics of the signal in preprocessing. Moreover, this approach directly employs high spectral resolution of group delay functions for accurate detection of onsets.

This method is chosen based on two observations; First, the percussive signals in Carnatic music resembles AM-FM signals. Second, the envelopes of the signal can be used for GD based processing, in similar lines of (Shanmugam and Murthy, 2014*a*). The algorithm is evaluated on a large dataset of $\approx 17,000$ strokes consisting of *tani avarthanams* as well as ensemble of recordings (multiple instruments in a concert).

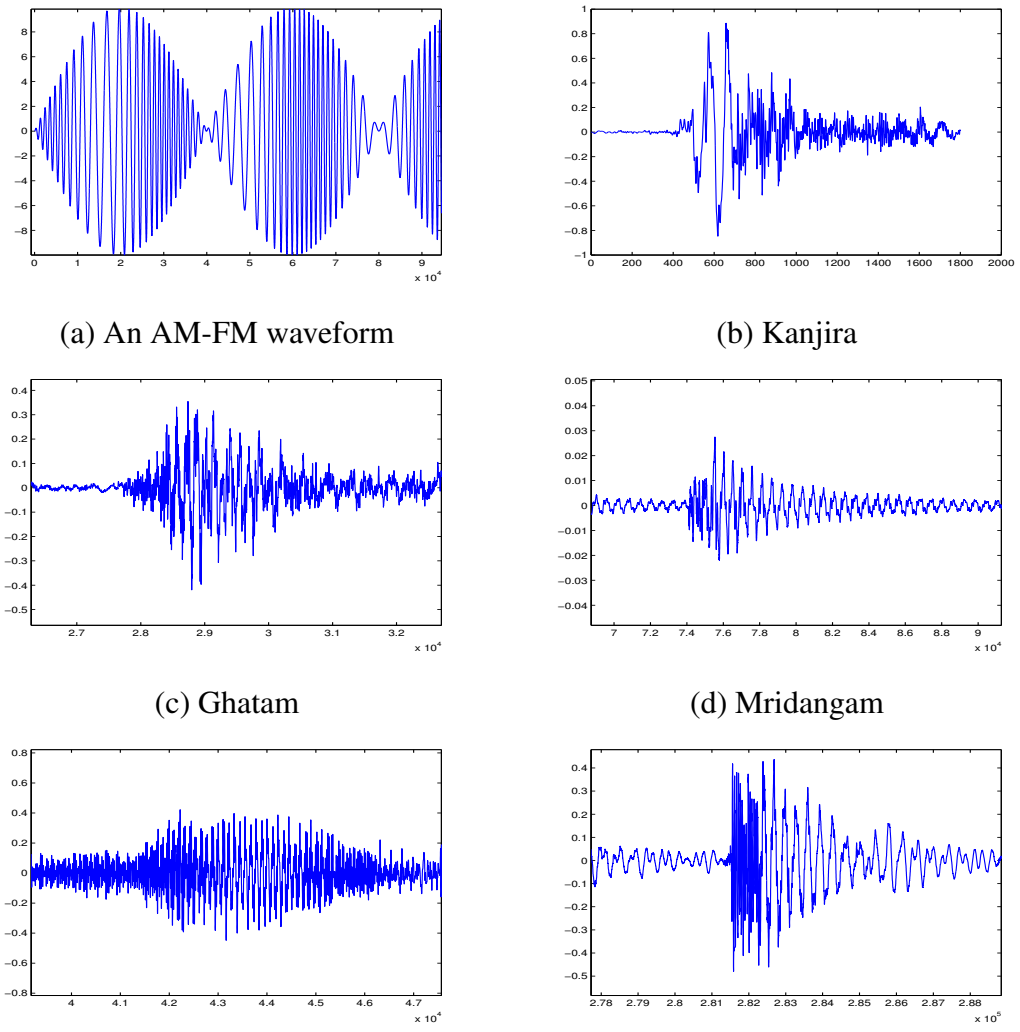
Amplitude and frequency modulation (AM-FM)

In the context of communications, a message signal $m(t)$ is modulated with a high frequency carrier signal $s(t)$ before transmission. The primary purpose for this is the reduction in antenna size for the transmitter and receiver, and also to multiplex different signals. Various modulation schemes exist, and are characterised by the influence of the message signal ($m(t)$) on the carrier ($s(t)$) signal. In the case of amplitude-frequency modulation (AM-FM), both the amplitude and frequency of the carrier signal are influenced by message signals $m_1(t)$ and $m_2(t)$. Figure 4.9 (a) presents an example using sinusoids in place of $m_1(t)$ and $m_2(t)$.

It is observed that most of the percussive strokes in Carnatic music can be modelled by an AM-FM signal, based on the variations in amplitude and frequency at the vicinity of an onset. Figure 4.9 illustrates this similarity by comparing a carrier signal modu-

[1]Collaborative work with Manoj Kumar P. A. (Kumar, 2015)

lated using sinusoidal message signals with individual strokes of mridangam, ghatam, kanjira, morsing and thavil. It is proposed that, the messages $m_1(t)$ and $m_2(t)$ contain information necessary to pinpoint the location of onsets. A demodulation technique is essential to extract this information before proceeding to locate the onsets.



(a) An AM-FM waveform

(b) Kanjira

(c) Ghatam

(d) Mridangam

(e) Morsing

(f) Thavil

Figure 4.9: Resemblance of Carnatic percussion strokes to an Amplitude-Frequency modulated (AM-FM) waveform.

Demodulation and Envelope Detection

Consider a signal $x(t)$ that is amplitude-frequency modulated. The basic representation is given as:

$$x(t) = m_1(t)cos(\omega_c t + k_f \int m_2(t)dt) \tag{4.9}$$

$m_1(t)$, $m_2(t)$ represent the message signals, k_f is the frequency modulation constant and ω_c is the carrier frequency. Differentiating $x(t)$ with respect to time,

$$x'(t) \approx -e(t)sin(\omega_c t + k_f \int m_2(t)dt) \tag{4.10}$$

where

$$e(t) = m_1(t)(\omega_c + k_f m_2(t)) \tag{4.11}$$

The term $m_1'(t)cos(\omega_c t + k_f \int m_2(t)dt)$ has been ignored in Equation 4.10 since ω_c can be assumed large. Both the messages now become part of the amplitude in Equation 4.10. All the information about an onset is argued to be contained within the envelope function $e(t)$. $e(t)$ is extracted from $x'(t)$ using the Hilbert transform as follows:

Any real-valued signal $S(t)$ with Fourier transform $S(w)$ can be represented by its analytic version (as introduced by Gabor in 1946 (Gabor, 1946)) and is given by

$$S_a(t) = 2\int_0^\infty S(w)exp(j2\pi wt)dw \tag{4.12}$$

Hence, it is the inverse Fourier transform of positive frequency part alone. In terms of input signal $S(t)$,

$$S_a(t) = S(t) + iS_H(t) \tag{4.13}$$

where, $S_H(t)$ is the Hilbert Transform of $S(t)$. Real part of an analytical signal represents the actual signal and imaginary part is it's Hilbert Transform. Magnitude of the analytical signal gives an estimate of the envelope.

Minimum phase group delay processing

The high resolution (HR) property of group delay is exploited for estimating the onset times from the music signal. The signal is converted to minimum phase by using the modification discussed in Section 2.3.1. The envelope function $e(t)$ defined in Equation 4.11 is considered as one half of the magnitude spectrum for a hypothetical signal $s(t)$. The assumption is valid since $e(t)$ is a positive function. The minimum-phase group

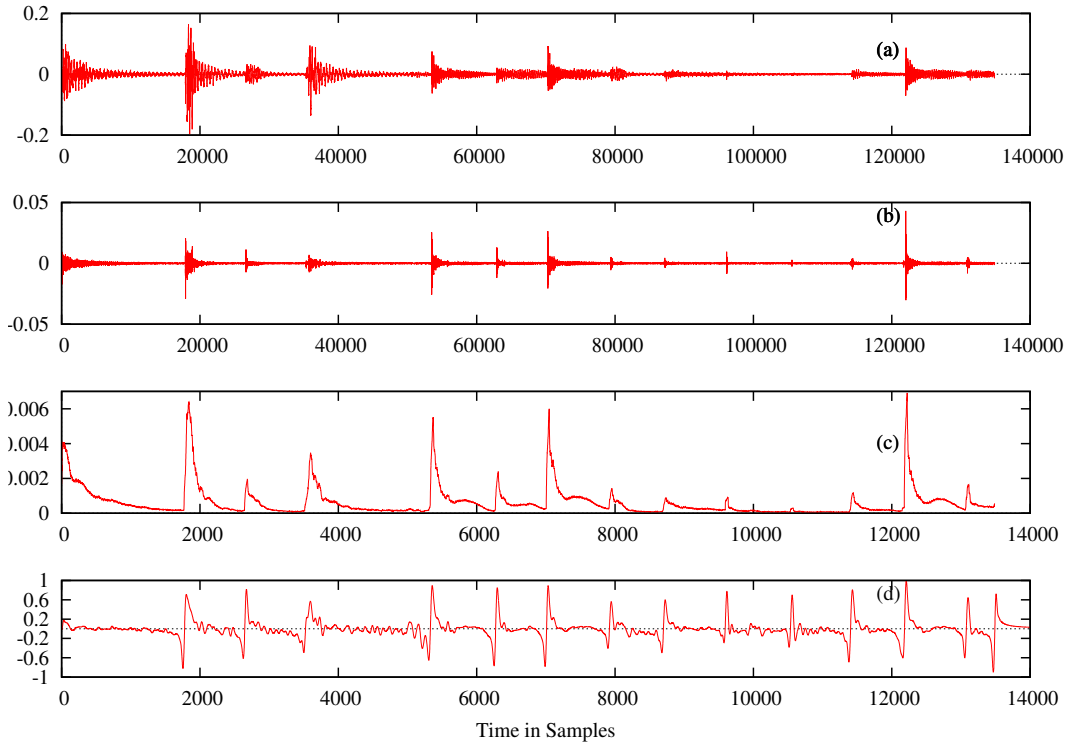


Figure 4.10: Working of proposed algorithm. (a) Music signal, (b) derivative of music signal, (c) envelope estimated suing Hilbert transform, and (d) minimum phase group delay computed on the envelope.

delay processing of the signal results in GD domain estimates and is considered as the *detection function* in this approach. Please refer to Section 2.3.1 for details about minimum-phase GD processing.

Figure 4.10 illustrates the various stages of the algorithm using a mridangam clip with both loud and silent strokes. Differentiating the music signal highlights the location of all onsets considerably (b) when compared to the original music signal (a). The envelope function is estimated on (b) using Hilbert transform and downsampled. Treating (c) as the positive half of the magnitude spectrum of a hypothetical signal, the minimum phase group delay equivalent is computed in (d). It is interesting to note in the final step that the group delay function emphasises all strokes approximately to an equal amplitude irrespective of the original amplitudes in the music signal. Onsets are reported as instants of significant rise using a threshold.

4.3.6 Experimentation

Since there is no existing annotated dataset for Indian instruments, one dataset is created in the course of this work. The experiments have been performed on this dataset of annotated onsets, consisting of $17,592$ strokes from mridangam, ghatam, kanjira, morsing, thavil and an ensemble of all instruments except thavil. The recordings were taken from *tani avarthanams* (solo) of Carnatic concerts. The mridangam recordings are split into musically relevant 'phrases' by professional musicians while all other instruments are split into segments of 20s each. The entire dataset is sampled at 44.1KHz, and the combined duration of the 277 clips is ≈ 42 minutes. Instrument-wise details of the dataset are provided in Table 4.3.

Table 4.3: Instrument wise details of datasets used in percussive onset detection.

Instrument	**Total length (min:sec)**	**Strokes**
Mridangam	18:41	5982
Ghatam	4:14	2616
Kanjira	3:11	1377
Morsing	6:35	2184
Thavil	4:39	2904
Ensemble	5:00	2529

The closely spaced onsets are not merged unlike (Eyben *et al.*, 2010; Böck *et al.*, 2012) while calculating the F-measure since it becomes impossible to differentiate between simple and composite[2] strokes, the latter being quite common in mridangam, kanjira and thavil. Further, the cases of multiple onset outputs within the threshold of a target, and a single onset output within the threshold of various targets are not considered. They are treated as false positives and false negatives respectively.

The comparison is made with a state-of-art algorithm (Schlüter and Böck, 2014) based on CNNs. The optimum results are reported by varying the respective thresholds in the proposed algorithm as well as in (Schlüter and Böck, 2014).

Results

The comparison between the proposed and CNN based approach is given in Table 4.4. Both onset detection algorithms report fairly good F-measures on all instruments, which

[2]Composite refers to both left-right strokes occurring together in mridangam, and strokes which involve multiple fingers for kanjira and morsing

Table 4.4: Performance measures for the proposed algorithm vs Convolutional Neural Networks based state-of-art algorithm on Carnatic percussion instruments.

	Group delay			CNN		
Instrument	**Precision**	**Recall**	**F measure**	**Precision**	**Recall**	**F measure**
Mridangam	0.972	0.974	0.973	0.964	0.941	0.952
Ghatam	0.968	0.924	0.946	0.968	0.943	0.956
Kanjira	0.936	0.914	0.925	0.98	0.972	0.976
Morsing	0.925	0.907	0.916	0.936	0.937	0.937
Thavil	0.95	0.805	0.872	0.991	0.82	0.897
Ensemble	0.927	0.886	0.906	0.956	0.921	0.938

is expected of percussion instruments. The results reported are over number of strokes ranging from 1377 to 5982. Hence, an improvement of even 1% absolute is significant as it contains multiple (>14) strokes which are correctly identified. It is interesting to note from Table 4.4 that the proposed algorithm stands out in performance for the mridangam dataset, in spite of significant variations in tempo and loudness. The lower recall for the proposed algorithm on all other instruments can be attributed to the fast tempo in comparison to the mridangam, suggesting even better temporal resolution might be necessary. Both algorithms report significantly low recall on kanjira and thavil, which have a relatively high appearance of composite strokes. Finally, morsing strokes lack a sharp onset, and this is directly reflected in the decrease in F-measures. A purely signal processing based approach is proposed using the segmentation property of group delay functions, that is comparable in performance with the state-of-art technique. It is important to mention that CNN based approach does not scale when the datasets are not used for training the model. A reasonably large annotated dataset for onset detection is created consisting of only Carnatic instruments as a part of this work.

4.3.7 Conclusion to percussive onset detection

We present the relevance of GD in estimating the time-events for a musical signal in this task. A purely signal processing-based approach is proposed using the segmentation property of group delay functions, that is comparable in performance with the state-of-the-art technique. Detection of the beginning of a musical stroke is an essential step in stroke classification and meter detection in MIR. We have also created a fairly large annotated dataset for onset detection consisting of only Carnatic instruments in

the course of this work.

4.4 Spike Estimation from Neuronal Signals

This section intends to estimate the time-events of spike occurrence by exploiting the high-resolution property of group delay. Several models and algorithms have been proposed for spike estimation task over the past decade. Nevertheless, it is still hard to achieve accurate spike positions from the Ca^{2+} fluorescence signals. While the existing methods rely on data-driven methods and the physiology of neurons for modelling the spiking process, this work exploits the nature of the fluorescence responses to spikes using signal processing. The Ca^{2+} indicator responds to a spike with a sudden rise, that is followed by an exponential decay. The Ca^{2+} signal is interpreted as the response of an impulse train to the change in Ca^{2+} concentration, where the Ca^{2+} response corresponds to a resonator. Minimum-phase group delay-based filtering approach is applied on the Ca^{2+} signal for resolving the spike locations.

4.4.1 Introduction to spike estimation

Spike estimation from calcium (Ca^{2+}) fluorescence signals is a fundamental and challenging problem in neuroscience domain. Neurons generate spikes which carry the information in the brain. Any neuroscience analysis with cognition requires the information about this spiking process. The generation of spikes can be considered as a time-event. Underlying calcium concentration change owing to a spike can be fetched via fluorescence signals.

The time-scale of time-events in this task varies from milliseconds to several seconds. This also depends on the calcium indicator used for obtaining the fluorescence signal. This section further corroborate the high-resolution property of group delay as the analysis is carried out based on the possible bandwidth ranges at which GD exhibits this property. We consider bandwidths of various indicators and indeed observe that it is possible to apply GD processing for fluorescence signals. In addition, this section also proposes an end-to-end encoder decoder neural network-based approach for spike estimation from neuronal signals. It establishes that each of the time-event detection stages can be realised in a supervised setup using convolutional, dense and transposed con-

volution layers. A detailed analysis of the training targets, architecture, generalization ability etc. are also provided in this section.

4.4.2 Background

Neurons generate spikes which encode stimulus information in the brain. Billions of neurons and their numerous task-specific activations are fundamental to any cognitive/behavioural activity. The spikes obtained from the neurons in local brain circuitry are essential for higher-level retrieval tasks. The information processing in and via neurons is of interest to the research community. The temporal location or time of occurrence of spikes and its rate carry information about the about the underlying stimuli.

The spike positions can be obtained using electrophysiology or by imaging techniques. In electrophysiology, micro-electrodes attached to each of the neurons are used to get the action potentials. However, this measurement is not only limited by poor spatial resolution but is also an invasive and expensive setup. In widely used two-photon imaging technique, the mouse is either genetically encoded or injected with Ca^{2+} indicators having fluorescence emitting capability (Stosiek *et al.*, 2003). The activities of a population of neurons are recorded via latest imaging techniques (Cotton *et al.*, 2013; Grewe *et al.*, 2010; Pachitariu *et al.*, 2018). These noninvasive techniques are less harmful to animal compared to the invasive neurophysiology.

The Ca^{2+} fluorescence signals are very noisy with quick rise times and long decay tails. The information about the underlying spiking process is masked in the fluorescence signal. The fluorescence owing to a spike decays gradually and interferes with the adjacent spikes. A slowly varying signal with a reduced temporal resolution is thus generated. The actual neuronal action potentials need to be extracted from these noisy signals for performing efficient neuronal processing. The following reasons limit the spike estimation task: The background fluorescence varies with time and is indistinguishable from the fluorescence changes during spike occurrence. The two-photon imaging and the fluorescence signals are contaminated by random noise. Finally, the transients at intra-cellular Ca^{2+} level have large time constants, and several such transients are non-linearly added resulting in poor tracking capabilities especially when spikes overlap (Akerboom *et al.*, 2012; Vogelstein *et al.*, 2009; Chen *et al.*, 2013*a*; Wilt *et al.*, 2013). Hence, it is essential to understand and de-noise these fluorescence traces

to get the actual spike estimates.

Existing algorithms

Several methods proposed for inferring the spiking information can be categorised into generative approaches and supervised methods.

1. Generative methods model the fluorescence signal as the responses of the indicator to the changes in calcium concentration which in-turn are caused by spikes. They rely on several model-specific assumptions. Deconvolution-based approaches are based on convolutive assumptions about the spiking process (Yaksi and Friedrich, 2006; Vogelstein *et al.*, 2010; Pnevmatikakis *et al.*, 2016) whereas biophysical model-based approaches estimate the most probable spike train which generated the fluorescence output (Deneux *et al.*, 2016). Other model-based approaches include template matching (Greenberg *et al.*, 2000; Greenberg *et al.*, 2008; Grewe *et al.*, 2010), likelihood-based alignment, auto-regressive formulation (Friedrich and Paninski, 2016) and approximate Bayesian inference based on de-convolution (Vogelstein *et al.*, 2009, 2010; Pnevmatikakis *et al.*, 2013). These models are limited by the a priori assumptions about the model which has stringent approximations regarding the shape of the calcium response and statistics of noise.

2. Supervised models predict the spike information from the fluorescence traces either using a set of features derived from the signal or using the raw-signal. Data-driven methods are gaining traction recently owing to the availability of simultaneous electrophysiological and two-photon scanning-based recordings of neurons. The supervised model discussed in (Theis *et al.*, 2016) is used for learning the λ parameter of a given Poisson model using a neural network (Theis *et al.*, 2013). Recent methods use fluorescence signals with or without a contextual window (supplementary material - (Berens *et al.*, 2018)) for estimating the spike information. Neural network-based variants such as "convi6", "Deepspike", "Purgatorio', and "Embedding of CNNs" had varied levels of success and outperformed data-driven baseline method (Theis *et al.*, 2016) on a standard evaluation framework (supplementary material- (Berens *et al.*, 2018)). A gated recurrent unit (GRU)-based approach recently attempted to estimate action potentials directly from the 2D calcium imaging output, combining regions-of-interest (ROI) and spike estimation tasks (Linsley *et al.*, 2018).

Deneux et. al. (Deneux *et al.*, 2016) rely on the physiology of neuronal firing using a non-linear model which estimates the spikes based on the most-likely spike train given a fluorescence recording. Parameters of this algorithm are auto-calibrated for optimal performance at the expense of significant computational cost.

The popular Vogelstein algorithm (Vogelstein *et al.*, 2010) attempts to de-convolve the noisy fluorescence signal to obtain a calcium trace from which the spike positions are estimated. Other signal processing algorithms in the literature are inferior to the MLspike algorithm in terms of performance metrics (Deneux *et al.*, 2016; Pachitariu *et al.*, 2018; Berens *et al.*, 2018). Machine learning-based methods such as STM (Theis *et al.*, 2016) and other learning techniques (Berens *et al.*, 2018; Pachitariu *et al.*, 2016; Speiser *et al.*, 2017) need sufficient fluorescence examples along-with the ground truth information for training the system and often have a limited performance on unseen datasets. STM is a model-dependent technique where the spikes are modelled by a Poisson distribution (Theis *et al.*, 2016; Kass and Ventura, 2001). The parameter λ of the distribution is then learned using a neural network (Theis *et al.*, 2013).

MLspike (Deneux *et al.*, 2016) is the best signal processing algorithm (Berens *et al.*, 2018) and CNN-LSTM model introduced in (Berens *et al.*, 2018) is the state-of-the-art supervised algorithm for the spike estimation task.

Evaluation metrics

Performance evaluation of a spike estimation algorithm should consider the overall shape of the spike information signal, the accuracy of the time estimates of spikes with various thresholds. Measures used in the literature vary depending on the algorithm. However, spikefinder challenge (Berens *et al.*, 2018) standardised the evaluation scheme in which three evaluation measures are considered. A single metric for spike inference is debatable as it only reveals limited parts of the spike inference (Kümmerer *et al.*, 2017). Theis Theis *et al.* (2016) considered correlation, F-measure and information gain whereas the challenge considered correlation, rank and area under the receiver operating Characteristics (AUROC or AUC). Commonly used evaluation measures for spike estimation are explained below.

1. Correlation

Correlation measures the similarity of the two signals by considering the overall shape of the spike information signal. It is the most commonly used evaluation measure in the literature (Deneux *et al.*, 2016; Theis *et al.*, 2016) and the primary evaluation measure used in the spikefinder challenge. In this metric, between every sample of the original and the estimated spike information, the Pearson correlation coefficient is calculated. Following the standard framework (Deneux *et al.*, 2016; Theis *et al.*, 2016; Berens *et al.*, 2018), a bin-width of 40 ms is used for obtaining the correlation measure. The correlation values cannot be interpreted directly as the spike rates or counts as the correlation measure does not consider the uncertainty of the predictions (Theis *et al.*, 2016).

2. Area Under the ROC Curve (AUC)

The AUC is the area covered between the true positive rate (TPR) and the false positive rate (FPR) in a Receiver Operating Characteristic (ROC) and is also used as a measure in spikefinder challenge (Berens *et al.*, 2018). The area is obtained by changing the threshold of the spike information signal for selecting the appropriate threshold or operating point. AUC is a measure of how well a parameter can distinguish between positive and negative instances. This metric counts all the peaks higher than the parameter as a spike, and hence the changes in the relative height at different temporal positions are not considered. AUC is not a good measure for methods that directly results in the binary spike train.

3. F-measure

The harmonic mean of sensitivity and precision is called as F-measure. Precision is the measure of relevance of the selected spikes. Sensitivity is the ratio of true spike positions detected. For this measure, the input needs to be in a discrete format. The distance between the predicted spikes and the ground truth is computed using a dynamic programming algorithm (Victor and Purpura, 1996) which penalises the distance for insertions, deletions, and shifts of spikes (Deneux *et al.*, 2016). Hence, F-measure is a measure of the exactness of the spike with respect to the ground truth at a high-resolution (up to 10 ms resolution).

4. Rank

Spearman's rank correlation coefficient is used as the secondary evaluation metric in spikefinder challenge (Berens *et al.*, 2018). It measures both the strength and the direction of the association between two ranked variables. The ranking is based on the dynamic ranges of the spike information and discrete spike train. This correlation measure also considers the possible non-linear relationship between the variables.

5. Information Gain

Information Gain is a measure proposed in (Theis *et al.*, 2016). This measure provides a model-based estimate of the amount of information about the spike train extracted from the calcium trace. It takes into account the uncertainty of the predictions. Assuming an average firing rate of λ and a predicted firing rate of λ_t at time t, the expected information gain (in bits per bin) can be computed as,

$$I_g = \frac{1}{T}\sum_t log_2\frac{\lambda_t}{\lambda} + \lambda - \frac{1}{T}\sum_t \lambda_t \tag{4.14}$$

assuming Poisson statistics and independence of spike counts in different bins. The estimated information gain is bounded from above by the (unknown) amount of information about the spike train contained in the calcium trace, as well as by the marginal

entropy of the spike train. The relative information gain is obtained by dividing the information gain averaged over all cells by the average estimated entropy. This can be interpreted as the fraction of entropy in the data explained away by the model (measured in percent points).

4.4.3 Motivation

Spike estimation can be interpreted as a peak estimation problem constrained by the nature of neuronal firing. The objective of any signal-based approach is to annihilate the baseline variations and other residuals and to enhance the peaks. Our method for spike estimation can be loosely-related to syllable segmentation from speech signals (Shanmugam and Murthy, 2014*a*), and estimation of onsets from that of music signals after converting it to an envelope function (Kumar *et al.*, 2015). However, the number of spikes occurring at a time is not linearly related to the magnitude of the Ca^{2+} fluorescence signal, and the nature of spike firing is unpredictable in general.

This subsection establishes that group delay functions can be exploited for the analysis of Ca^{2+} signals. The mathematical model for the Ca^{2+} signal is briefly discussed based on the observation from (Deneux *et al.*, 2016). Given the properties of Ca^{2+} signals and group delay functions, an attempt is made to establish a correspondence between Ca^{2+} signals and GD functions. The group delay-based processing step does not use domain information. It is primarily a filtering step that enhances the peaks of the fluorescence signal owing to the high-resolution property of GD functions. The properties of the Ca^{2+} signal are similar to that of a cascade of resonators. This property is used to illustrate the benefit of using group delay for spike estimation. This is of great significance for any peak picking task, and especially for Ca^{2+} fluorescence signals wherein the superposition of responses leads to hardly resolvable peaks.

Ca^{2+} fluorescence signals represent the neuronal activity over a period of time. This time series signal contains the spike time information embedded within the signal. However, as it is corrupted by noise, the exact response of Ca^{2+} to a spike and the fluorescence protein to Ca^{2+} is unknown and obtaining the spike timing of these signals is a cumbersome task. The changes of the short-term energy envelope for each syllable in speech and changes in the derivative envelope for each onset in music are similar to the fluorescence changes which correspond to the neuronal firing, although the duration of each of these events is different. Signal units consisting of onset, attack and decay with their respective orders is shown in Figure 4.11. Figure 4.11 (a) shows *"Va"*, a Hindi speech syllable 4.11 (b) shows *"Tha"*, a percussive stroke in Mridangam and 4.11 (c) shows a Ca^{2+} fluorescence segment generated due to an action potential. The bottom panel shows the minimum phase GD functions extracted from the corresponding signals in the top panel. In speech signals, a syllable is the smallest meaningful production unit. Similarly, for a percussion instrument, a stroke is the fundamental pro-

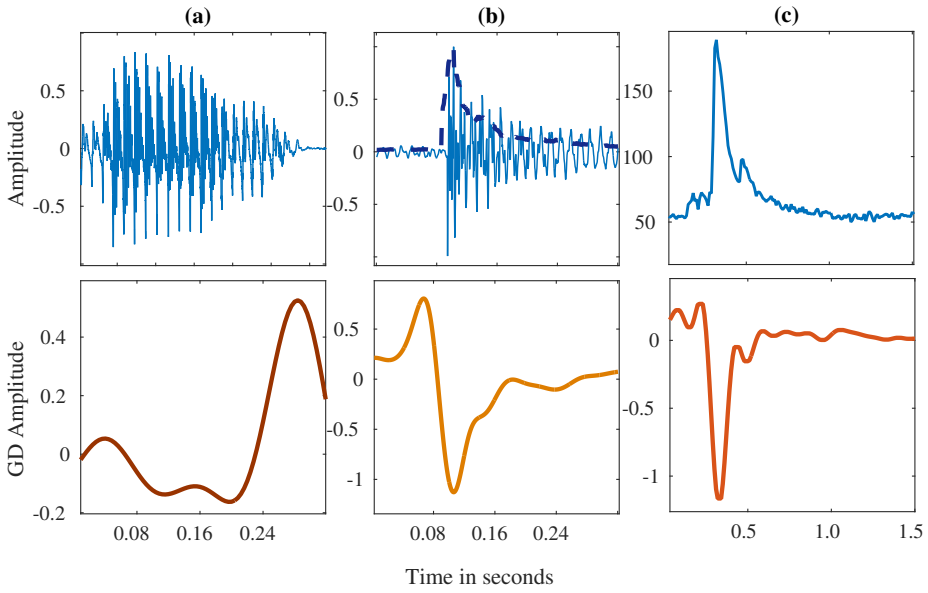


Figure 4.11: Motivation for Group Delay (GD)-based spike estimation. Figure shows the signal units having an onset, an attack, and a decay and its corresponding group delay domain representation.

duction unit. In neuronal signals, an action potential is a fundamental unit which can be observed as a peak in the Ca^{2+} signals. Depending on the kinetics of Ca^{2+} binding to the indicator dye or protein, this peak can have various rise and decay time constants. All these signals have an onset, an attack, and a decay. However, it is different from audio signal processing in the following aspects: the spectrum is richer and has several frequency components with variable characteristics across time, whereas the envelope of the audio signal is adequate to determine the syllable boundaries in speech, the sampling frequency is much lower in Ca^{2+} signals, and the dynamic range of input vary significantly for Ca^{2+} signals.

As shown in Figure 4.11, GD-processing leads to sharper peak locations for these tasks. The bottom panel (b) and (c) shows GD, and (a) shows inverse of GD as the valleys are more important than peaks in syllable segmentation task. The Ca^{2+} concentration change owing to a single spike and the resulting fluorescence change can be thought of as a single pole system. The responses to a set of spikes are added together in the time domain, and their Fourier representation should also be an addition of individual responses. The Ca^{2+} fluorescence signal is interpreted as a Fourier representation of addition of responses to individual spikes. The impulse response of a two-pole system shown in Figure 3.4 in Section 3.5.2 can be interpreted as the response of Ca^{2+} fluorescence to two independent spikes. It is the impulse response of a two pole system with an exponential decay for each of the poles for practical bandwidths. Though this interpretation considers a very simple model, it enables us to understand the working of group delay for spike estimation task. It should be noted that the model is used only

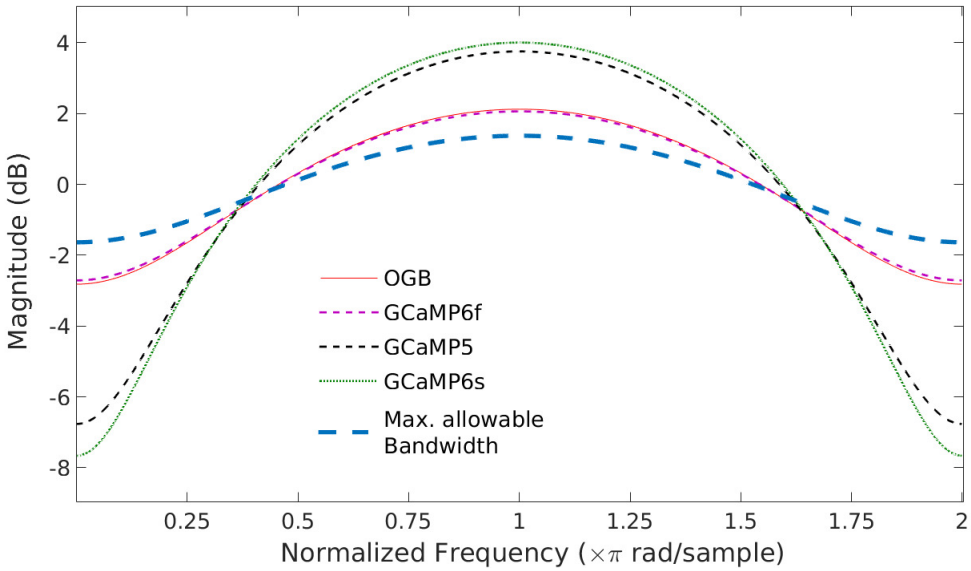


Figure 4.12: Frequency domain decay interpretation for various Ca^{2+} indicators.

for estimating the indicator bandwidths and thereby interpreting the Ca^{2+} signals as a one-sided magnitude spectrum.

4.4.4 GD for spike estimation

As discussed in the beginning of this Section (4.4.2), most of the existing algorithms are inspired from the neuro-physiology of the Ca^{2+} responses. In this thesis, the nature of the Ca^{2+} fluorescence signals is studied in detail, and the same is related to the group delay algorithm. The Ca^{2+} signal can be interpreted as the response of a decaying exponential system to a train of impulses. The decay rate of the exponential varies with the Ca^{2+} indicator. Pure signal processing makes it possible to apply this algorithm as an enhancing step to the existing algorithms and it is illustrated for the MLspike approach. It is suitable to apply this algorithm in an online-fashion as it is an unsupervised algorithm.

4.4.3.1 GD interpretation of Ca^{2+} signals

A non-model-based signal processing algorithm, known as GDspike, is proposed for spike estimation from Ca^{2+} fluorescence signals. The approach relies on the ability of the group delay function to resolve closely-spaced spikes. The fluorescence signal is considered as the positive frequency half of the magnitude spectrum. This assumption of Ca^{2+} signals makes it similar to a formant structure in speech signal with a centre frequency and bandwidth. This is justified by (a) the correspondence between the Ca^{2+} decay and practical single pole bandwidth and (b) the feasibility of group delay-based smoothing on this magnitude spectrum. Minimum phase group delay representation

not only amplifies the sharp and tiny fluorescence changes but also restricts the peaks for large fluorescence changes. Since the Ca^{2+} fluorescence signal is interpreted as a magnitude spectrum representation of the addition of responses to several spikes, the fluorescence decay time-constant should be in a resolvable range in the frequency domain. This section considers different time constants for the decay, and show that group delay analysis indeed provides a high resolving capability.

The decay of Ca^{2+} concentration with respect to time after the occurrence of a spike is exponential in nature. This is modelled in (Deneux *et al.*, 2016) as:

$$c_t = e^{-\Delta t/\tau} c_{t-1} + n_t \tag{4.15}$$

where, n_t is the number of spikes between time $t-1$ and t and this time difference is given by Δt. The exponential decay of Ca^{2+} concentration depends on the time constant (τ) of the indicator being used. The fluorescence change owing to the change in Ca^{2+} concentration is modelled as a non-linear function of Ca^{2+} concentration c_t. The change in calcium fluorescence in the time interval Δt is given by ΔF. Hence, the fluorescence change (ΔF) will be slower than the corresponding Ca^{2+} concentration change. The ΔF thus has a larger time constant than the Ca^{2+} decaying time constant. The decay time-constant varies based on the indicator. Table 4.5 gives the time constants for each of the indicators as obtained from manual calibration (Supplementary material from (Deneux *et al.*, 2016)) and their corresponding bandwidths in the frequency domain. The quantity σ is directly proportional to the bandwidth of the pole location $e^{-\sigma+j\omega}$. The value of $e^{-\sigma}$ denotes closeness of the pole to the unit circle and is inversely proportional to the bandwidth.

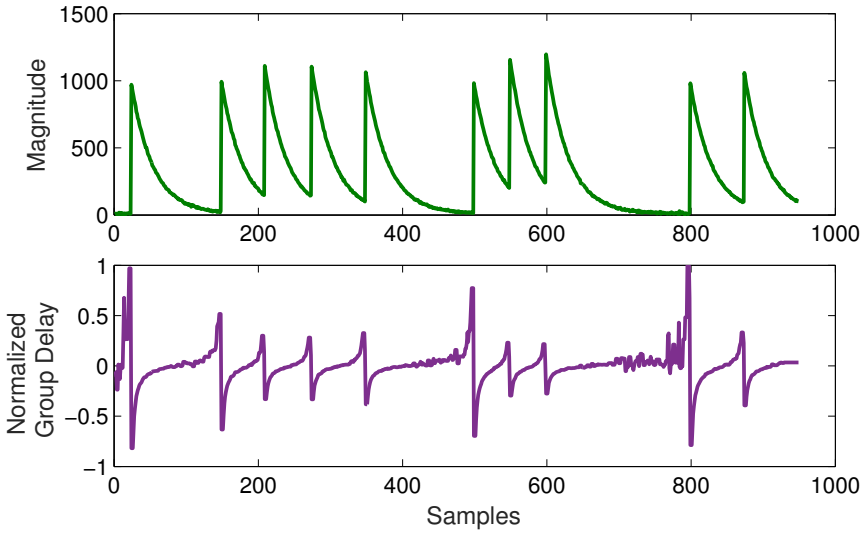


Figure 4.13: Group delay representation of a set of exponential with instantaneous rise time.

Table 4.5: Time constants and the corresponding $e^{-\sigma}$ for various Ca^{2+} indicators.

Measure	OGB	GCaMP5	GCaMP6s	GCaMP6f
τ	0.78s ($\pm$ 0.37)	1.63s ($\pm$ 0.55)	1.87s ($\pm$ 0.35)	0.76s ($\pm$ 0.17)
$e^{-\sigma}$	0.277	0.541	0.586	0.268

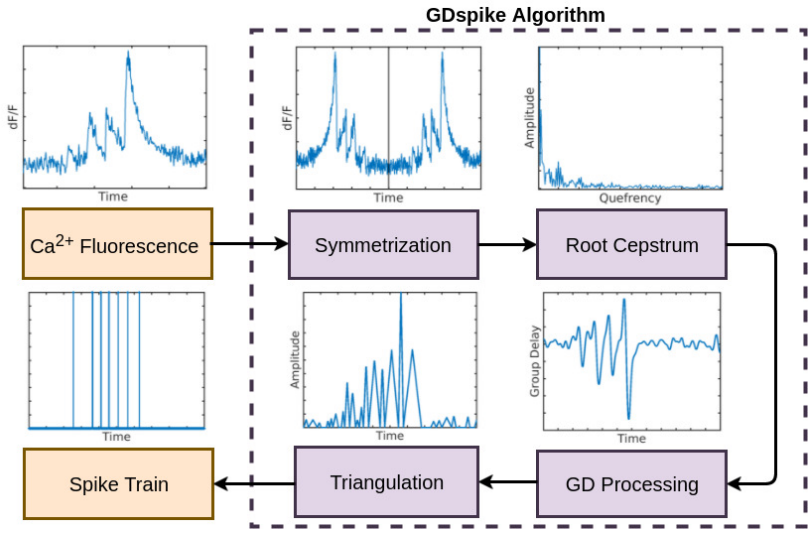


Figure 4.14: Block diagram of the GDspike approach.

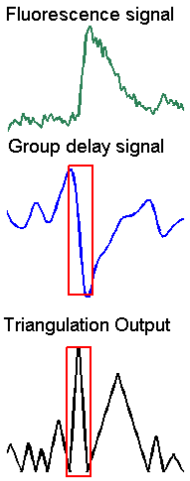


Figure 4.15: Triangulation step.

The $e^{-\sigma}$ value obtained for the fastest indicator is 0.268. The bandwidth of a pole location $e^{-\sigma+j\omega}$ is obtained by its correspondence with the decaying exponential. This suggests that the decay rate for any of the indicators is no more than 0.76s. Group delay processing will work provided these decay rates correspond to the bandwidths for which the group delay function has a better resolution. In single pole analysis in Section 3.4, it was observed that the high resolution property holds good when $e^{-\sigma} \in [N-1/N+1, 1]$. For practical bandwidth considerations (3 dB), $e^{-\sigma} \in [0.1715, 1]$. In the context of the indicators, it can be seen that the bandwidth (or rather the decay rate) lies in the interval [0.26,0.59], a subset of the interval obtained in Section 3.4. Hence, the input signal can be considered as the magnitude spectrum and the application of group delay processing is justified. Using the decay rate as bandwidth, the spectra of the resonances corresponding to that of the various indicators is plotted in Figure 4.12.

When a neuron is in an excited state, a spike is observed. The response of an indicator bonded with Ca^{2+} ions to a spike can be thought of as a convolution of an impulse to the indicator. It has a magnitude response similar to a single pole system as the convolution with impulse does not affect the magnitude. The observed Ca^{2+} fluorescence at a neuron, at its simplest representation, is nothing but the superposition of response of the indicator to a train of impulses. Figure 4.13 shows a synthetic Ca^{2+} signal obtained as the convolution of impulse train and impulse response of the Ca^{2+} concentration (Equation 4.15). Here, a simple exponential model with zero rise time (onset and attack at same temporal position) is considered for the impulse response of calcium

concentration. For each of the spike positions, the corresponding bandwidth representation ($e^{-\sigma}$ = 0.90) is lower than the maximum allowable bandwidth ($e^{-\sigma}$ = 0.17) in the magnitude spectrum which makes the group delay analysis feasible. The bottom panel shows the corresponding group delay representation which is sharper than the magnitude spectrum. This interpretation considers the nature of group delay response to an ideal condition, not considering the baseline variations or noise. Nevertheless, it is effective as the aim of this observation is to provide the intuition about the group delay processing for spike estimation. The bandwidth interpretation is valid for indicators with non-zero rise times as well since the Ca^{2+} fluorescence signal is modelled as a magnitude spectrum.

4.4.3.2 GDspike algorithm

Figure 4.14 shows the block diagram of the proposed approach. The Ca^{2+} signal which encodes the spike information is the input to the algorithm. Instead of estimating the group delay function from the Ca^{2+} signal, the signal is considered as a magnitude spectrum owing to two reasons; First, it is a positive function and second, the decay rate matches with the required bandwidths for group delay-based smoothing operation. A minimum-phase equivalent GD function is then obtained for this magnitude spectrum (Yegnanarayana and Murthy, 1992) by taking the causal portion of the inverse Fourier transform of the Ca^{2+} signal, after making it symmetric with respect to the magnitude axis. It is observed that the locations of the spikes are characterised by a transition from positive to negative. A triangulation step is performed to make all the peaks positive. Mid-point of every high-to-low change of the signal in the group delay domain is considered as a peak, and the saddle points (where the first order difference is zero) are regarded as the zero positions, as shown in Figure 4.15. These positions are connected to form isosceles triangles of various heights. The red boxes indicate the transition and the corresponding isosceles triangle. It should be noted that no threshold is needed to get this spike information.

Group delay-based processing step does not use any domain information and is a filtering step to enhance the peaks of the Ca^{2+} signal. Most algorithms in the literature process the fluorescence signal to obtain a signal in which the spike locations are enhanced. In our algorithm, the signal is considered as a spectral signal. Hence, it is regarded as the one-sided magnitude spectrum of a hypothetical signal (Figure 4.16(a)). This assumption is valid not only as the decay range falls in the practical bandwidth for high resolving capability, but also as the hypothetical signal is minimum phase which makes group delay processing feasible. The group delay signal has the information about the spike position. This is refined using triangle approximation step to obtain the detection function (Figure 4.16(c)). Observe that spike information (c) captures the smallest of the changes in the fluorescence signal corresponding to a spike position.

This detection function is similar to the spiking probabilities or spiking information.

Algorithm 1 *GDspike*

Input: Ca^{2+} Fluorescence signal $C[n]$.
Output: Spike information signal $Sp[n]$.

1: Consider the fluorescence signal, $C[n]$ = positive side of absolute of $(F.T\{h[n]\})$.
2: Calculate the hypothetical signal, $h[n] = F^{-1}(C[n] + C[-n])$.
3: Take $h[n], n > 0$. This is limited by window scale factor $(winscale)$, an empirical parameter[3]. $h_1[n]$ = $(\frac{length(h[n])}{winscale})$.
4: Compute group delay, $GD[n]$ = Group delay of minimum-phase signal, $h_1[n]$ using Equation 2.6.
5: Triangulation step: Find the zero crossing positions i in $GD^{'}[n]$ $(\forall i \in N)$ and compute $Sp[\frac{2i+1}{2}] = abs(GD[i] - GD[i+1])$ and $Sp[i] = 0, \forall i$.

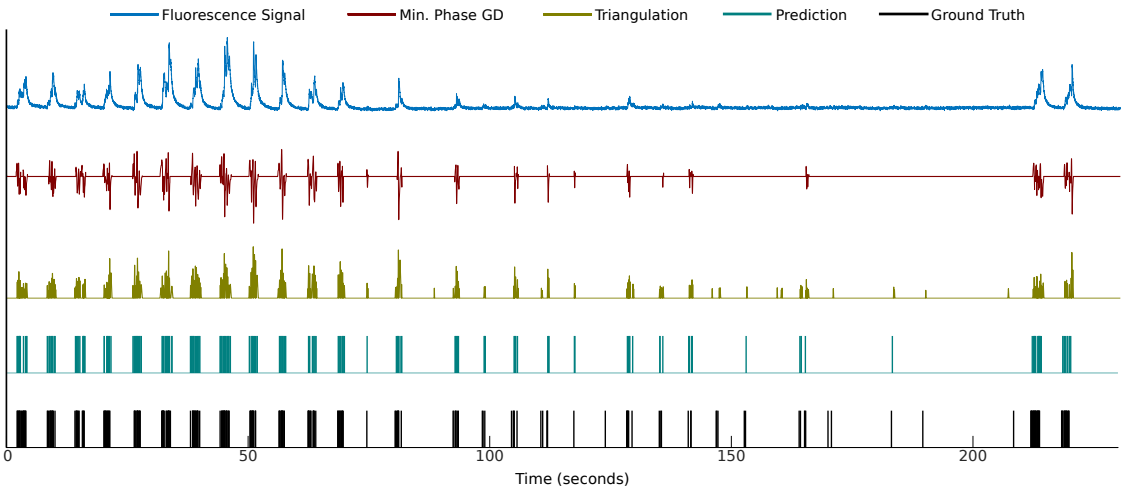


Figure 4.16: GDspike Algorithm. (a) A segment of a fluorescence signal, (b) its minimum phase group delay representation, (c) the spike information, (d) predicted spike train, and (d) ground truth.

4.4.3.3 GDspike as a post-processing step

As GD is actually agnostic to the signal, it can be used as a post-processing step in the existing algorithms with spike information as the output. These signals are less-noisy in nature and are correlated more with the actual spike train than the fluorescence signal. Use of GD to post-process these signals enables sharper spike locations at the output. This work shows that the GDspike also can be used as a post-processing step for the best performing signal-processing method (in spikefinder challenge) for Oregon Green Bapta (OGB) data: MLspike, resulting in better spike detection. GDspike performs consistently well for GCaMP indicator-based fluorescence signals. The group delay based post-processing step is therefore not required for these datasets (Sl No. 1-5 in Table 4.6). GDspike is applied as the post-processing method on the MLspike algorithm for the OGB datasets. The spiking probabilities from the MLspike model act as the

[3]Uniformly across various domains, it has been observed that a WSF of 4 works

input signal to the GDspike algorithm, which sharpens the envelope and provides a more accurate spike information.

4.4.5 Experimental procedure

4.4.4.1 Data collection procedure

The datasets for experimentation are obtained both from publicly available datasets and from the authors of MLspike. The procedure for collection of the fluorescence signals uses two-photon imaging technique. Cells are labeled either with a virus carrying a genetically encoded calcium indicator (GECI), or with a calcium indicator dye such as oregon green bapta (OGB) injected into the cortex. The calcium sensor enables neurons to show fluorescence changes, due to variations in calcium concentration with each action potential. The fluorescence change is captured, and a sequence of images consisting of a population of neurons is obtained. This is reported in detail in (Deneux *et al.*, 2016). The experiments are repeated across trials. Electrophysiology recordings are done simultaneously to obtain the ground truth for evaluating the performance. Different scanning methods leads to various sampling rates of the Ca^{2+} fluorescence imaging. Data is acquired using either acousto-optic or galvanometric scanning methods. The dataset includes multiple brain areas (visual cortex, hippocampus, and barrel cortex) and different species of animals (rats and mice).

4.4.4.2 Datasets used

A publicly available dataset[4] contributed by Svoboda lab, at Janelia Research Campus (Akerboom *et al.*, 2012; Chen *et al.*, 2013*b*; Dana *et al.*, 2016)) and four other datasets (Deneux *et al.*, 2016) are used to evaluate GDspike. The spikefinder challenge (Berens *et al.*, 2018) had some GCaMP and OGB datasets. Spikefinder challenge signals are pre-processed to remove linear trends. The unprocessed raw signal is used for analysis. It is a larger dataset as compared to the spikefinder evaluation dataset. The datasets 1-5 (Table 4.6) are identical to the ones optionally used for training in the spikefinder contest, though pre-processing is not performed. The datasets are chosen such that they correspond to different areas of the brain, different experimental setup and, fluorescence colours. In total, 9 test datasets are considered in comparison to 5 datasets in the spikefinder challenge.

The details of the datasets are given in Table 4.6. The first set of data (Sl No. 1-5 in Table 4.6) are collected from the neurons of *in vivo* mice visual cortex and use GECI proteins. Other datasets are collected using OGB indicators (Sl No. 6-8 in Table 4.6) and also include mouse *in vitro* and *awake* set up. The ground truth is recorded at a very

[4] `http://crcns.org`

Table 4.6: Dataset used for evaluation (S.R.: sampling rate (in Hz)).

No.	# cells	Indicator	System	S.R.	Spikes	Ref.
1	9	GCaMP6s	Mouse visual cortex	60	2123	(Chen *et al.*, 2013*b*)
2	11	GCaMP6f	Mouse visual cortex	60	4536	(Chen *et al.*, 2013*b*)
3	9	GCaMP5k	Mouse visual cortex	50	2735	(Akerboom *et al.*, 2012)
4	11	jRGECO1a	Mouse visual cortex (Red)	25	9080	(Dana *et al.*, 2016)
5	10	jRCaMP1a	Mouse visual cortex (Red)	15	3624	(Dana *et al.*, 2016)
6	10	OGB (Weizmann)	Rat barrel cortex	25	901	(Theis *et al.*, 2016)
7	13	OGB (Marselie)	Rat barrel cortex	50	5241	(Theis *et al.*, 2016)
8	1	OGB (invitro)	Mouse hippocampus (invitro)	25	74	(Theis *et al.*, 2016)
9	2	GCaMP6s (Budapest)	Mouse visual cortex (*awake*)	50	140	(Theis *et al.*, 2016)

high sampling rate $\simeq 10$ kHz compared to the slowly varying Ca^{2+} fluorescence signal which is recorded at a sampling rate, ranging from 15 Hz to 100 Hz via two-photon imaging. Following the protocol used in spikefinder challenge (Berens *et al.*, 2018), the signals are re-sampled to a uniform sampling frequency of 100 Hz, and the evaluations are performed at 25 Hz (equivalent to a bin width of 40 ms) on the spike information signal.

4.4.4.3 Baseline algorithms

The algorithms used for comparison vary from supervised (Theis *et al.*, 2016), models based on physiology (Deneux *et al.*, 2016) and de-convolution-based (Vogelstein *et al.*, 2010) approaches. The comparison is made with the most popular algorithm (Vogelstein), best signal processing (MLspike), and the baseline (STM) algorithms of spikefinder challenge (Berens *et al.*, 2018). Supervised baseline method (STM) uses Poisson distribution for modelling the spike information whereas MLspike uses a biophysical model consisting of a noise and baseline fluorescence modelling to estimate the most likely spike train from the Ca^{2+} imaging data. The original version of the algorithm presented in (Deneux *et al.*, 2016) is used for post-processing. The Vogelstein de-convolution algorithm is a popular signal processing algorithm for spike estimation. It has been shown in (Theis *et al.*, 2016) that the STM method outperforms other methods such as Peeling (Grewe *et al.*, 2010), Sequential Monte-Carlo (Vogelstein *et al.*, 2009), constrained de-convolution (Pnevmatikakis *et al.*, 2016), and other algorithms used for comparison in (Theis *et al.*, 2016).

4.4.4.4 Evaluation metrics

A bin-width of 40 ms is used for obtaining the correlation, following the standard framework (Deneux *et al.*, 2016; Berens *et al.*, 2018; Theis *et al.*, 2016). The correlation values cannot be interpreted directly as the spike rates or counts as the correlation measure does not consider the uncertainty of the predictions (Theis *et al.*, 2016). Therefore this work rely on other measures as well. The signal obtained after triangulation step is used as the spike information for computing the AUC. It is calculated on the spike

information signal at 40 ms bin width (equivalent to 25Hz).

F-measure is calculated between every sample (at 10ms bin width) based on the protocol used in (Theis *et al.*, 2016). For computing the F-measure, the output is converted to discrete spikes by thresholding the spike information signal. The discrete spike train output of the MLspike algorithm is used for calculating this measure. For Vogelstein and STM, an optimum global threshold is experimentally chosen for the given datasets. Since these algorithms were designed to generate spike information, the results are reported on the best-possible threshold values. For GDspike, the signal is thresholded at a global value computed based on the mean and standard deviation of the spike information signal averaged over all datasets (*not* the best possible threshold), although GDspike is also an algorithm designed to generate the spike information. This threshold is used for comparison with the baselines. The F-measure is also computed using dataset-wise thresholds, (a) *dataset-wise 1*: by 3-fold cross-validation on 60% of each dataset to determine the threshold X (Mean + X $\times$ Standard Deviation) and then testing on held out data (40%) and (b) *dataset-wise 2*: by using 20% of the data to determine the threshold (with maximum F-measure) and testing on 80% of the data (averaged over 5 random sets).

4.4.6 Results and analysis

Table 4.8 shows the average performance of the GDspike in comparison with the baselines. It outperforms the Vogelstein algorithm for all of the evaluation metrics. It also has second-best average AUC. It has better correlation & AUC than MLspike. MLspike had the best correlation in spikefinder challenge. This might be because of (a) using the a-posteriori probabilities rather than maximum-a-posteriori spike trains and (b) 5 or 6 parameter tweaking based on the training datasets (supplementary material of (Berens *et al.*, 2018)). This provides better results than auto-calibration, but the parameters needed to be tuned for each training dataset, in the similar lines of (Pachitariu *et al.*, 2018).

The correlation of spike information with the ground truth is limited by the nature of the continuous output in both GDspike and MLspike. Our "de-convolved-trace" is the triangulation output. The peak values of triangulation output are well-suited to estimate a discrete spike train as the AUC measure suggests. However, the shape is very different from the sharp spiking probabilities, and this makes the correlation worse for GDspike. The authors of MLspike also state that its "estimation accuracy is ranked inferior to that of the other algorithms when quantified using correlation". The average correlation between discrete spike train and ground truth is observed to be 0.349 for MLspike and 0.262 for GDspike, suggesting that the discrete spike train is important for these algorithms. Multiple measures are therefore used to evaluate algorithms to ensure that the pros and cons of an algorithm are reflected. It should be noted that as this

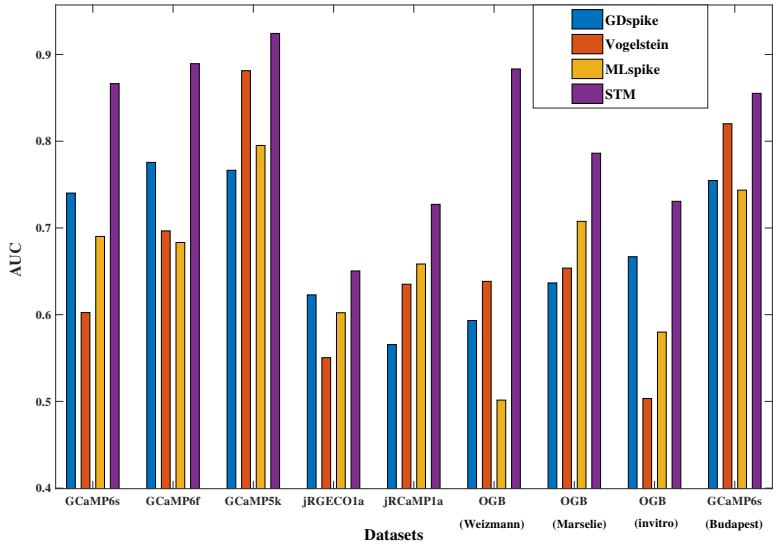


Figure 4.17: Dataset-wise ROC for various approaches.

work is not modelling the spiking process, the time delay between the spike occurrence and the corresponding fluorescence change (Chen *et al.*, 2013*b*) is not captured in the GDspike algorithm. This delay is not very relevant for GENIE dataset (Pachitariu *et al.*, 2018).

Dataset-wise results

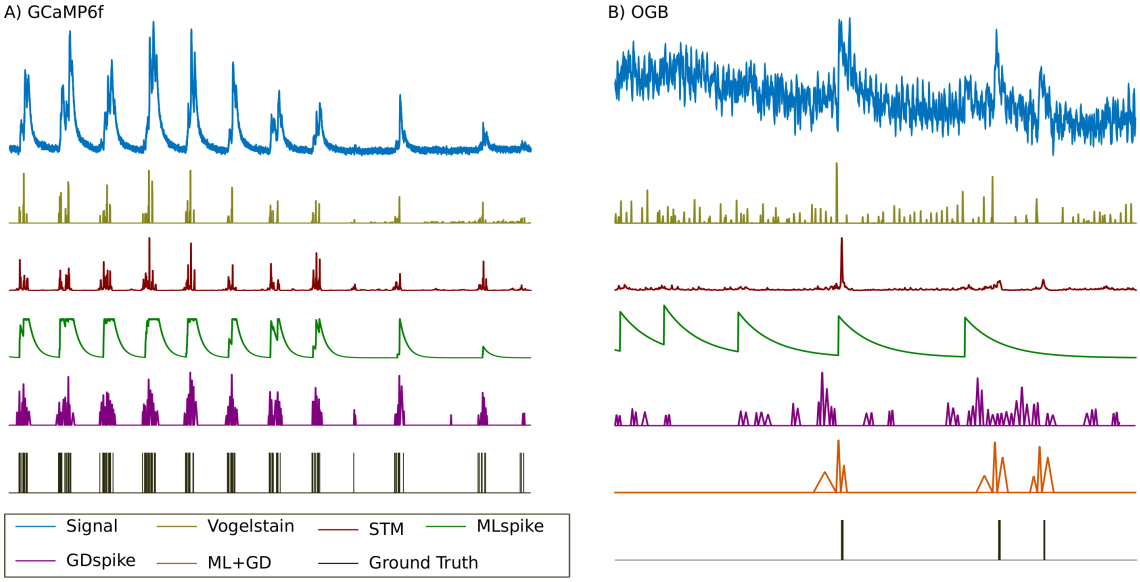


Figure 4.18: Examples of spike information obtained by GDspike and the baselines with (a) GCaMP6f indicator and (b) OGB indicator.

Table 4.9 shows the result obtained by GDspike on various datasets (averaged over all the trials and cells). The F-measure and correlation are better for green GCaMP indicators which are slowly varying and, less-noisy, hinting that the spike information is

easier to threshold. Figure 4.17 shows the Area Under ROC. AUC measure is consistent across the datasets. Figure 4.18 shows the spike information obtained by GDspike and baseline algorithms on representative examples with GECI and OGB indicators.

Dataset-wise F-measure for each algorithm is shown in Figure 4.19 (*left*). The minimum F-measure obtained is higher for MLspike and GDspike in comparison with Vogelstein and STM. The F-measure is not computed on the optimal threshold, rather on a simple rule-based global threshold obtained by computing the mean and standard deviation. The dataset-wise thresholding (shown as dataset-wise 1 and dataset-wise 2 in Table 4.7 and discussed in Subsection 4.4.5) provides an an improved F-measure. However, to have an unbiased comparison, the discrete spikes obtained using the global threshold is used for comparison with optimally-thresholded baselines.

Table 4.7: F-measure with the dataset-wise thresholds (on the test set).

Dataset	**Global**	**Dataset-wise 1**	**Dataset-wise 2**
GCaMP6s	0.58	0.60	0.53
GCaMP6f	0.52	0.52	0.56
GCaMP5k	0.38	0.41	0.39
jRGECO1a	0.32	0.25	0.39
jRCaMP1a	0.29	0.28	0.38
OGB (Weizmann)	0.23	0.47	0.39
OGB (Marselie)	0.31	0.34	0.47
OGB (invitro)	0.36	0.59	0.51
GCaMP6s (Budapest)	0.56	0.65	0.47
Average	0.39	0.46	0.45

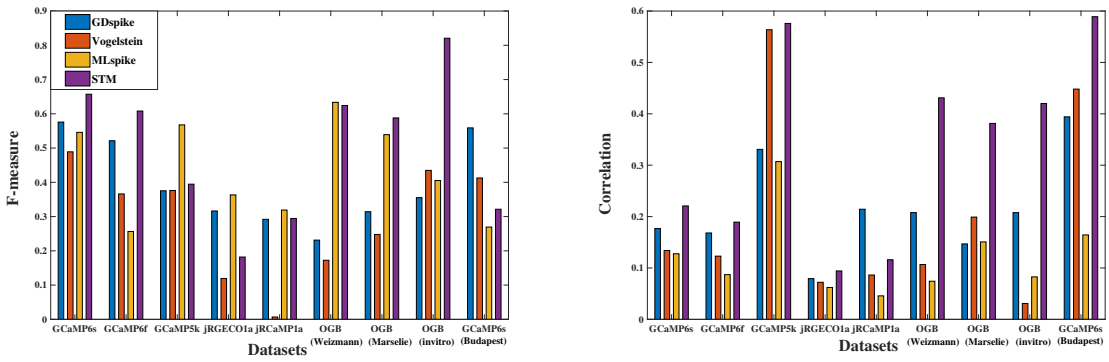


Figure 4.19: Comparison of algorithms based on (*left*) F-measure and (*right*) Correlation.

Figure 4.19 (*right*) shows the correlation measure. STM has the best correlation measure, and GDspike has the second best. The correlation of GDspike and MLspike becomes better when it is calculated on the discrete spike train. Thus the spike information signal is less-representative of the actual estimation ability of these algorithms. It is important to note that GDspike does not require complex modelling of baseline fluctuations and noise to perform the spike prediction.

Computational complexity

The runtime (in hh:mm:ss) for all the 9 dataset on an Intel(R) Core(TM) i7-4930K CPU @ 3.40GHz machine is 00:03:26 for Vogelstein, 00:02:30 for STM (Testing time), 00:04:34 for GDspike whereas it is 01:00:54 for MLspike. Hence, GDspike is 13 times faster than MLspike. This huge variation for MLspike is due to the auto-calibration of parameters. The calibration also requires a sufficient number of isolated spikes and becomes less accurate at high spiking rates (supplementary material- (Deneux *et al.*, 2016)).

STM method seems to provide the best results on average, though not consistent across the datasets. The STM approach performs well on the datasets similar to the training dataset but suffers when a dissimilar dataset to the training set is used for evaluation (dataset 4 and 5 in Table 4.6). OGB indicator captures the relative fluorescence amplitude change with respect to most of the spikes (unless they occur in bursts) as OGB is faster than GCaMP indicators. GDspike detects most of these spikes. However, it suffers from lower recall owing to false alarms for the OGB examples where the fluorescence signal is very noisy. This makes GDspike inferior to MLspike and STM for noisy OGB examples. Hence GDspike is used as a post-processing step, not as a stand-alone algorithm for the OGB data.

Table 4.8: Comparison with the baseline approaches.

Algorithm	Recall	Precision	F-measure	Correlation	AUC
Vogelstein	0.548	0.368	0.292	0.196	0.665
STM	0.762	0.426	0.499	0.335	0.813
MLspike	0.513	0.618	0.433	0.128	0.662
GDspike	0.413	0.687	0.393	0.214	0.680

Post-processing performance

Table 4.10 presents the results of the experiments in which GDspike is used as a post-processing step on the spiking probabilities obtained using MLspike. Observe that there is an improvement in performance on most of the metrics. MLspike modelling results in de-noised Ca^{2+} traces and GD enhances the resolution of the peaks. The combined approach makes use of the ability of MLspike to detect the precise locations and group delay converts it to a more resolved shape. This results in larger correlation and AUC measures. Figure 4.18(b) shows this improvement on an example OGB signal, where the *analog* output of MLspike is used as the input to GDspike. Observe that combined method picks only the actual spike positions, even if MLspike and GDspike cause false positives. Applying a single threshold on the triangulation output is not optimal and results in a decreased F-measure compared to MLspike in some cases.

Our experimental evaluation considered multiple evaluation measures, different sam-

pling rates of inputs, various brain regions, mouse conditions, indicators and, fluorescence colours. The proposed approach is better than Vogelstein with similar computational complexity. It is inferior to MLspike on the discrete spike train and superior on the spike information signal, with lesser time complexity. GDspike is agnostic to the signal and can be run in an online fashion.

Table 4.9: Performance of GDSpike for evaluation datasets.

Dataset	Recall	Precision	F-measure	Corr.	AUC
GCaMP6s	0.58	0.72	0.58	0.177	0.74
GCaMP6f	0.50	0.73	0.52	0.168	0.78
GCaMP5k	0.71	0.36	0.38	0.331	0.77
jRGECO1a	0.42	0.40	0.32	0.08	0.62
jRCaMP1a	0.26	0.63	0.29	0.214	0.57
OGB (Weizmann)	0.14	0.95	0.23	0.208	0.59
OGB (Marselie)	0.22	0.85	0.31	0.147	0.64
OGB (invitro)	0.20	0.94	0.36	0.208	0.67
GCaMP6s (Budapest)	0.69	0.60	0.56	0.394	0.76

Table 4.10: Performance of GDSpike as post-processing step for different OGB datasets.

Sl. No	Algorithm	Dataset	Recall	Prec.	F-measure	Corr.	AUC
1	GDspike	Weizman	0.14	0.95	0.23	0.21	0.59
	MLspike		0.90	0.51	0.63	0.07	0.50
	ML+GD		0.36	0.80	0.48	0.28	0.81
2	GDspike	Marselie	0.22	0.85	0.31	0.15	0.64
	MLspike		0.65	0.58	0.54	0.070	0.64
	ML+GD		0.43	0.78	0.48	0.16	0.73
3	GDspike	invitro	0.20	0.94	0.36	0.21	0.67
	MLspike		0.50	0.62	0.41	0.08	0.58
	ML+GD		0.36	0.71	0.64	0.24	0.67

4.4.7 Spike Estimation using S2S

End-to-end methods for various applications have been developed in speech and audio processing, image processing and neuroscience fields. However, a network for the signal to signal transformation in the context of neuronal signals is not yet available. This work proposes a Signal-to-signal data-driven neural network (S2S) for the spike estimation task. This method works on the raw-fluorescence signal in an end-to-end manner to generate the spike information signal. The convolutional layers and the multi-layer perceptron together learn the spike characteristics and generate the spike information. S2S network synthesises the spike information signal from the calcium fluorescence signals, contrary to *all* the machine learning approaches to spike inference which predict one output value for an input feature/signal (with optional window).

It is argued that such a spike estimation network can outperform existing approaches as it reconstructs the spike information for each sample of the input. This network differs from other sequence-to-sequence models in several aspects. First, the nature of the output signal (discrete spike estimates) is very different from the input signal (calcium fluorescence trace). Second, it performs automatic short-time processing through shared weights across the temporal axis. Third, it enables us to observe the output of each layer and frequency responses of both analysis and synthesis layers in the network. Finally, this method shows the applicability of the signal-to-signal network for processing neuronal signals.

The S2S method is compared with competitive algorithms in the spikefinder contest with the same dataset and evaluation procedure. S2S provides state-of-the-art results for spike estimation. A detailed comparison with the best performing supervised model in the spikefinder contest is made. The research questions regarding the reliability, generalisation ability, dependency on training targets and, design concerns of S2S are studied. A layer-wise analysis of the network is performed to provide an intuitive explanation for the learning.

The S2S network is inspired by the nature of the required output, which is also a temporal signal with spiking probabilities/information. The intuitive idea is as follows: The aim is to extract a signal which has the "intended" characteristics of a spike information signal. Hence, the focus here is towards creating a signal of desired characteristics which is very similar to the discrete spike train and not towards finding a spiking probability given a window (or context) of the input signal. The proposed network aims to predict the same number of samples as the input signal. This requires the use of synthesis layers which provides the same output shape as that of the input. This also provides automatic short-time processing through the use of time-distributed fully connected layer architectures which result in a shared weight matrix, similar to the ones in convolutional layers.

Figure 4.20 provides the block diagram of the proposed signal-to signal conversion network (S2S) for spike estimation task. It has a symmetrical structure with respect to the layer dimensions. S2S consists of a convolutional input layer, which operates over the entire signal length. The outputs are fed to a ReLU non-linearity. Unlike *sigmoid* and *tanh* activations, the ReLU units in the neural network do not saturate. This activation function also helps in learning a non-negative representation in the successive layers. A fully connected layer is implemented in a time-distributed fashion which works on every temporal block of outputs. The parameters are shared across time. The third part of the architecture consists of synthesis filter which produces the output shape similar to the input shape to the network owing to the transposed-convolution structure. Hence, spike information is obtained for a single recording in a signal-to-signal manner.

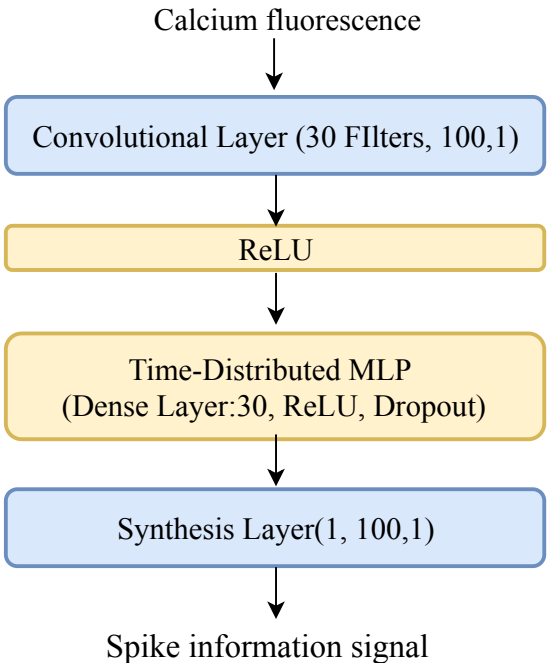


Figure 4.20: Block diagram of the proposed S2S for spike estimation.

4.4.8 Experimental Evaluation

Spikefinder challenge

Spikefinder challenge (`http://spikefinder.codeneuro.org/`) (Berens *et al.*, 2018) provided a new set of algorithms which performed better than the baseline (Theis *et al.*, 2016). These algorithms use different techniques, even-though their fusion could not further improve the results. The challenge was aimed to standardise the spike estimation evaluation and to provide the comparison of state-of-the-art techniques on the same datasets. Most of the top-performing algorithms in the contest used convolutional, recurrent and, deep neural networks and its variants (Berens *et al.*, 2018). All the top-performing data-driven algorithms have a recurrent layer in the network and hence are computationally complex. The best-performing supervised model used a convolutional neural network architecture with an intermediate LSTM layer to predict the spiking probability from a contextual window of fluorescence signal ("convi6" in the supplementary material of (Berens *et al.*, 2018)). Generative models such as MLspike (Deneux *et al.*, 2016) and (Friedrich and Paninski, 2016) performed comparable to the supervised approaches with a dataset-specific parameter tuning. MLspike uses a biophysical model and estimates the maximum probable spike information from the fluorescence signals. The second best generative approach in spikefinder is based on an auto-regressive approximation to the calcium fluorescence signal (Friedrich and Paninski, 2016). Spike information is then estimated by solving a non-negative sparse optimisation problem. Table 4.11 provide details of the methods proposed in the spikefinder challenge.

Table 4.11: Overview of top-performing algorithms in spikefinder challenge.

Team	Contributer(s)	new	type	Model/Architecture
Team1	T. Deneux	No	Generative	Biophysical model
Team2	N. Chenkov, T. McColgan	Yes	Supervised	conv / lstm
Team3	A. Speiser, J. Macke, S. Turaga	Yes	Supervised	RNN/CNN
Team4	P. Mineault	Yes	Supervised	residual / lstm
Team5	P. Rupprecht, S. Gerhard, R. W. Friedrich	Yes	Supervised	conv / max
Team6	J. Friedrich, L. Paninski	No	Generative	Autoregressive model
Baseline: STM	L. Theis	No	Supervised	DNN +Poisson model

The dataset released as a part of the spikefinder challenge (Berens *et al.*, 2018) is used for the evaluations. It includes five benchmarking datasets consisting of 92 recordings from 73 neurons. One part of this dataset was given for training the supervised models and the other part for testing as a part of the competition. Five datasets (Svoboda, 2015) from the GENIE project were also available for training the models. These additional datasets make sure that the supervised models do not over-fit to the training data. It further tests the generalisation capability of supervised methods. Other details about the spikefinder dataset are available at (Berens *et al.*, 2018). The evaluation measures considered are the Pearson correlation coefficient, Spearman's rank correlation coefficient and, the area under receiver operating characteristics (AUROC or ROC) in the order of preference. The computations are done at 25 Hz (40 ms bin width), enabling us to compare the performance with respect to spikefinder submissions. Performance of S2S is compared to top-six algorithms in the spikefinder contest. They are either based on generative (Deneux *et al.*, 2016; Friedrich and Paninski, 2016) or supervised (Theis *et al.*, 2016; Pachitariu *et al.*, 2018; Speiser *et al.*, 2017; Berens *et al.*, 2018) approaches. 7 out of top-10 algorithms are deep learning-based supervised algorithms.

Configuration of S2S

In the first layer, 30 convolutional filters (N_{filt}) of 1 sec width (W_{seq} equivalent to 100 samples) and sample by sample shift (W_{shift}=1) are used. The outputs of the convolution layer are passed through a ReLU non-linearity which ensures that the outputs of the filters are positive. The convolution results in a set of filter outputs having a length which is 1 sec lesser than the original, while maintaining the temporal order. The fully connected structure consists of a dense layer, followed by a ReLU and a dropout layer. Three fully connected structures, each of 30 hidden units takes the output from the ReLU and provide a linear transformation. It should be noted that the network is working at the 1-sec level, but maintains the temporal structure. THe synthesis filter at the final layer generates 1 sec signal for every input frame (100 samples from 30 (N_{filt}) filter outputs). This is done across the time steps as the weights are shared across the entire sequence. Weight sharing layers, automatic short-time processing, signal syn-

thesis are the major specialities of this network. The signal similarity measures could be used as an objective function since the conversion network is a signal-to-signal network. In this work, the Pearson correlation coefficient is used as the objective function to maximise during the training phase. The convolutional layer and every hidden layer is followed by dropouts.

S2S is very efficient owing to the small number of parameters. Each fully connected layer has 930 parameters (30×30+30 (for bias)), and analysis layer has 30 filters with 100 dimensions (30*100). There is only one synthesis filter of size 100. Hence, even for the most complex S2S, the number of parameters is 8,790. Lack of recurrent layers makes S2S training very fast. It is trained with Adam optimiser with cross-correlation as the objective function. The data is split into training (80%) and validation (20%) and train the network for 100 epochs with a batch size of 20. Early stopping with a patience factor of 6 is used to stop the training whenever the validation loss increases with respect to the previous epoch. This network is built using Keras (Chollet *et al.*, 2015) with TensorFlow (Abadi *et al.*, 2015) backend.

Results

The performance is compared with respect to the results reported in the spikefinder challenge (Berens *et al.*, 2018) (see Table 4.12). S2S network outperforms all the state-of-the-art methods. It improves the test correlation by 46% compared to the best performing algorithm in the spikefinder contest. Change in (Δ) correlation with respect to STM is very significant for the proposed approach (3.5 times compared to the best algorithm). It outperforms all supervised approaches. S2S provides an improvement of 56% for the secondary evaluation measure over the baseline with the best rank. It also has a comparable AUC with the best-baseline approach. The minimum deviation between the train and test correlation is observed for the S2S (0.0079). The performance of the proposed approach over the five different test datasets shows a commendable correlation with respect to the shape and monotonicity of the predicted spike information and ground truth. It performs equally well for both OGB and GCaMP indicators (see Table 4.13).

Table 4.12: Performance comparison of S2S with spikefinder baselines.

Team Name	Train correlation	Test correlation	Δ correlation	Rank	AUC
Team 1 MLspike new	0.4823	0.4382	0.0810	0.2878	0.846
Team 2 convi6	0.4727	0.4378	0.0806	0.3319	0.846
Team 3 DeepSpike	0.4730	0.4347	0.0775	0.3338	0.851
Team 4 Purgatorio	0.5370	0.4325	0.0753	0.3258	0.815
Team 5 Embedding of CNNs	0.4900	0.4291	0.0719	0.2822	0.821
Team 6 Suite2p	0.4752	0.4188	0.0617	0.3071	0.821
Baseline STM	0.4024	0.3572	-	0.2664	0.821
Proposed S2S	0.6325	**0.6404**	**0.2832**	**0.5208**	0.847

Table 4.13: Dataset-wise performance of S2S on the test set.

Measure	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5	Average
Correlation	0.657	0.910	0.585	0.440	0.611	0.640
Rank	0.420	0.816	0.525	0.261	0.581	0.521
AUC	0.901	0.901	0.874	0.682	0.879	0.847

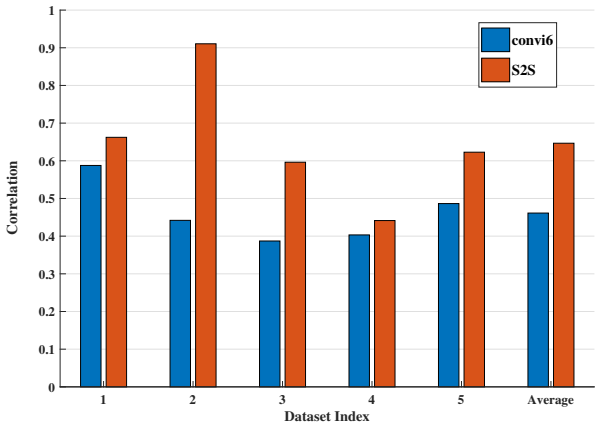


Figure 4.21: Comparisons of primary measure (correlation) with “convi6”.

The performance of S2S is compared with the best supervised approach (Team 2 in Table 4.12) in Figure 4.21 and Figure 4.22. S2S has better linear and rank correlation coefficients, and similar AUC in comparison to “convi6”. Signal conversion helps in the reconstruction of spike information that closely resembles the discrete ground truth. S2S has reliable output performance. i,e. It provides similar output after a fresh training process, independent of the initialisation of parameters. Best baseline performance (for primary evaluation measure) is achieved when the weights provided by the authors are used. The performance varies by 6% on average with respect to the baseline correlation when the model is trained from scratch with the same data and model configuration, but over two different training sessions. S2S is significantly faster than the baseline. While running on Sun GPU clusters, the S2S is found to be 50 times faster than the baseline model.

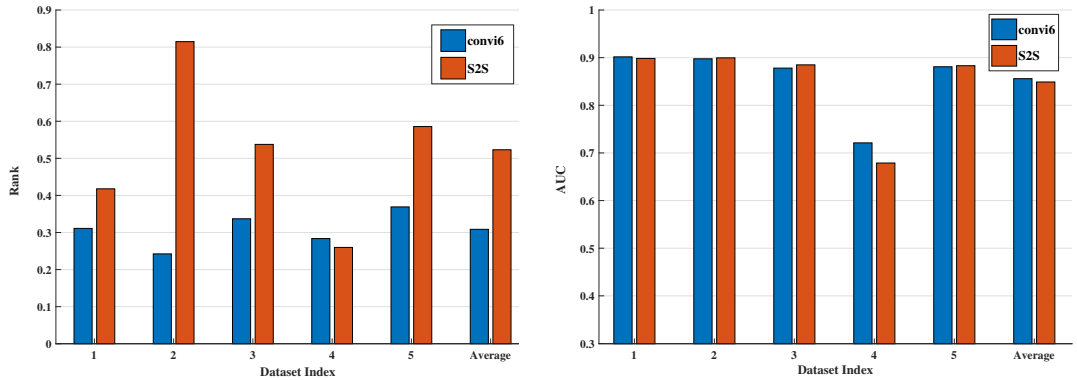


Figure 4.22: Comparisons of non-linear correlation and AUC with “convi6”.

Configuration

The system is evaluated first by considering the number of hidden layers which was varied empirically from zero to three. The three-layer network is found to discriminate the spikes well. S2S has similar measures to that of the baselines even without the dense layer (see Figure 4.23). When the number of hidden layers is increased to three, the performance is improved for *all* the measures. Increasing it beyond three does not improve the performance. Each hidden layer helps in denoising and improving the similarity of the layer output with respect to ground truth. Filters have improved denoising and spike-resolving capabilities for configuration without a hidden layer structure.

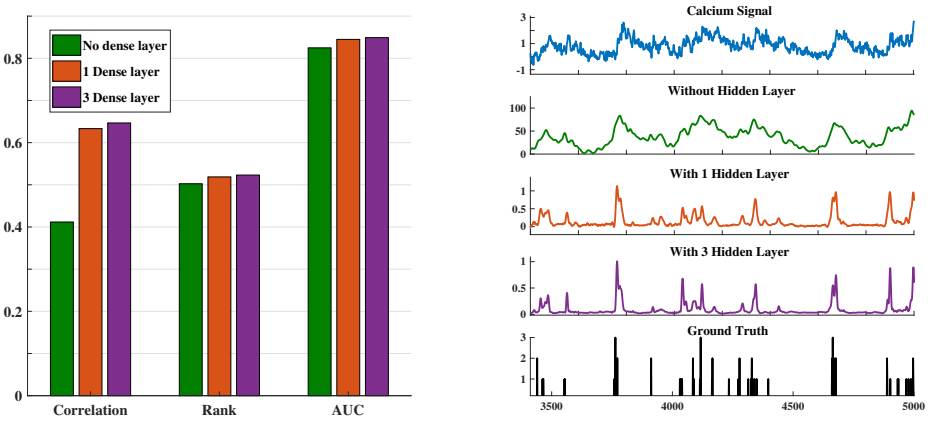


Figure 4.23: Performance with various number of hidden layers.

Training target

Since the ground truth is a sparse discrete signal, using discrete ground truth as the training target increases the training time and is often inefficient. This is true for supervised models predicting one output(mostly spiking probability) at a time. One clever way to surpass this is to use signals equivalent to the ground truth but are having lesser sparsity than the ground truth. For example, we could convolve the ground truth with exponential or Gaussian functions and make it less sparse which would help in efficient training. Being a signal-to-signal conversion approach, even training with the discrete ground truth as target yields an efficient system for the proposed approach. A less-sparse training target further improves the performance, and it also improved the training speed. Convolution of ground truth with a Gaussian window of variable size (see Figure 4.24) is considered. Improvement is noted for all the measures after this implementation. A Gaussian window of various sizes are considered, and it is observed that the performance is improved from (11,5) Gaussian window size to (33, 11) ((x, y) where, x= the number of samples and, y=standard deviation). Increasing the width of the Gaussian beyond this results in reduced performance since the shape of the training

target becomes very different from the discrete target (actual ground truth).

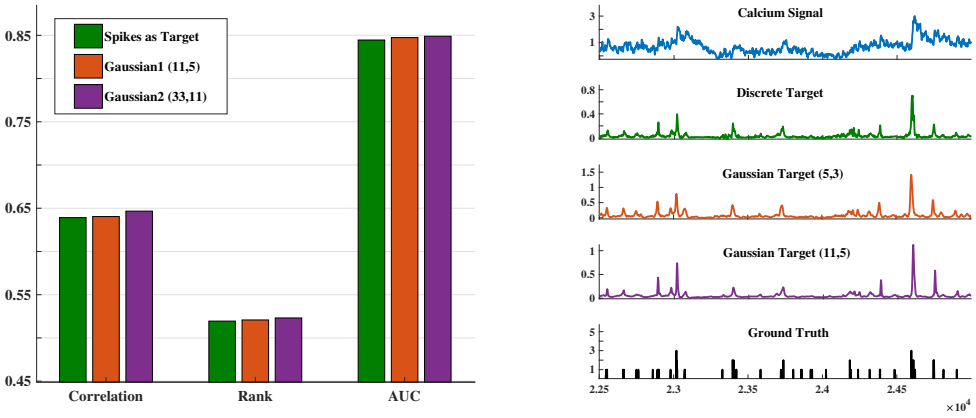


Figure 4.24: Performance with various training targets.

Generalisation ability

The network performance is compared with respect to the objective functions. The model trained with the correlation coefficient provides the best performance. Other loss functions such as mean squared error, cross-correlation with SoftMax non-linearity and their combinations fall short of Pearson correlation coefficient in terms of performance measures. However, the results are still superior with respect to the baselines. This refers to the learning ability of a synthesis network for spike estimation. The generalisation ability of the proposed approach is examined by training only with recordings from the GCaMP indicator and testing on the spikefinder test set which included three OGB recordings. This is performed on both one layer and three layer versions of S2S. The Gaussian window-based training target is used as it is found to be more efficient as discussed. Observe that (Table 4.14) for both 1 and 3 layer configurations, the model trained only with the GCaMP indicator has a competitive performance with that trained using all the training data. This indicates the ability of the network to generalise well on the unseen datasets. The evaluation measures on OGB datasets only gracefully degraded for the GCaMP model in comparison with the combined-model for *all* the datasets (Figure 4.25).

Table 4.14: Generalisation across indicators.

Configuration	Training Indicator(s)	Correlation	Rank	AUC
1 Dense layer	GCaMP	0.608	0.510	0.832
1 Dense layer	GCaMP + OGB	0.639	0.519	0.845
3 Dense layers	GCaMP	0.622	0.518	0.843
3 Dense layers	GCaMP +OGB	0.640	0.521	0.847

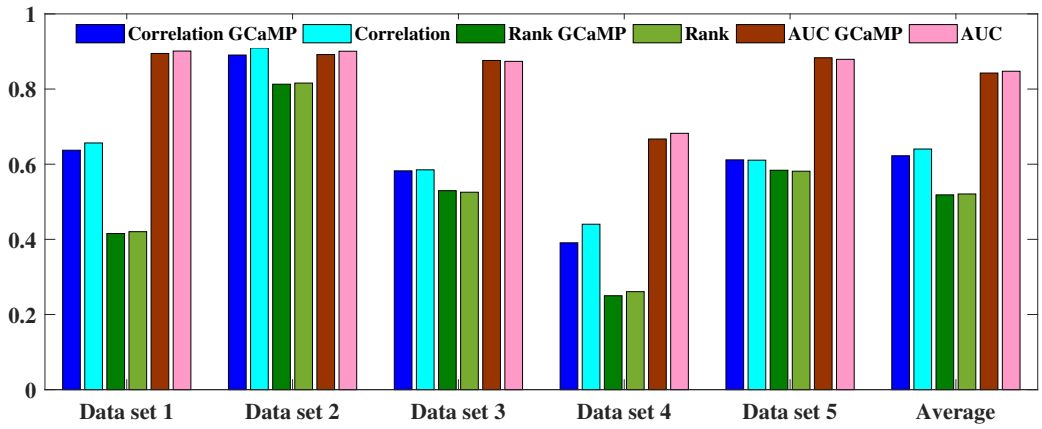


Figure 4.25: Dataset-wise performance showing the generalisation ability of S2S.

Layer-wise output

Figure 4.26 shows the layer-wise outputs for a 3 hidden layer configuration. For the given Calcium fluorescence input, the analysis layer seems to make it less-noisy while maintaining the possibly "required" information in the output. Observe that the shape of the input signal is preserved. This output becomes more discriminative at spike positions after passing through the first hidden layer and ReLU. This introduces some false information as well. The subsequent layers tend to make this discriminative information stronger while reducing the noisy (false) information. Synthesis layer creates the signal from the outputs of the fully connected layer. The filtering operation by this layer results in a more discriminative spike information signal having a good correlation with the ground truth. Hence, each layer is contributing towards the global goal of maximising the similarity between the spike estimates and the ground truth.

Filter responses

S2S learns to synthesise the spike information from the Calcium fluorescence signals owing to the presence of filters in the analysis and synthesis layers. The overall frequency response of the filters is computed (Figure 4.27). All the frequency components are equally responsive indicating that the filter layer acts as a pre-processor, unlike that of "convi6", which provides spike-related discriminative information. This variation might be due to the presence of a synthesis layer structure which is unique to S2S.

The cumulative frequency response of synthesis filter is very peculiar. They consist of responses across the entire frequency region, with a harmonic structure. This harmonic structure leads to the presence of a set of spikes in the cumulative impulse response of the synthesis filter (Figure 4.27). The synthesis filter provides a rich description of possible spike shapes and duration. A comparison across the architectures suggests that 3-CNN made the spike locations sharper, and closer to ground truth com-

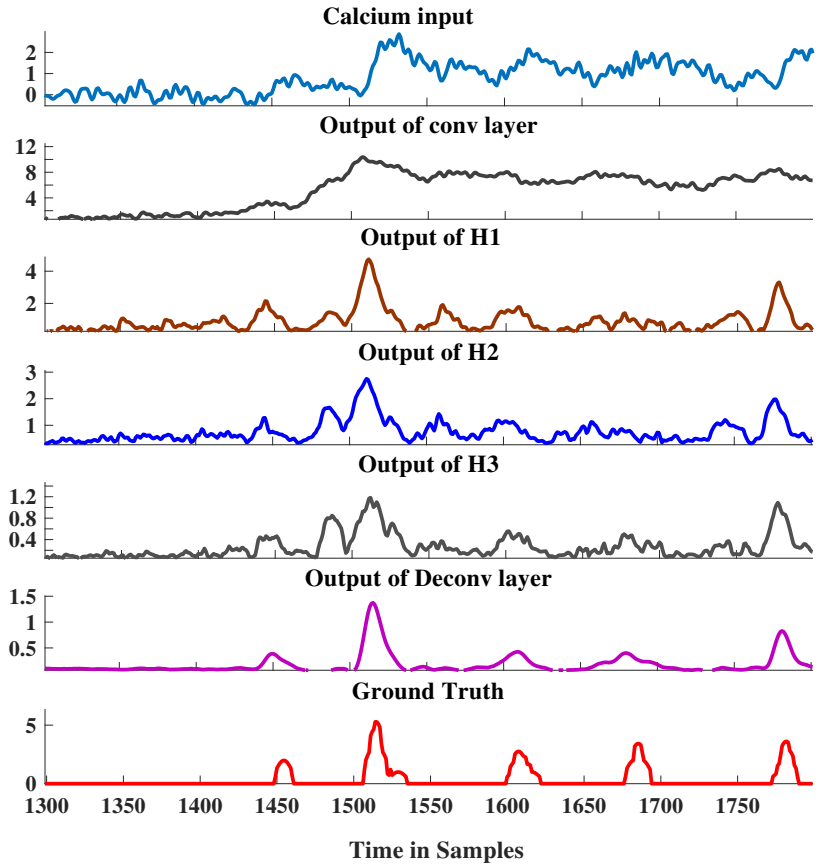


Figure 4.26: Layer-wise outputs of a 3-hidden layer S2S network.

pared to 1-CNN.

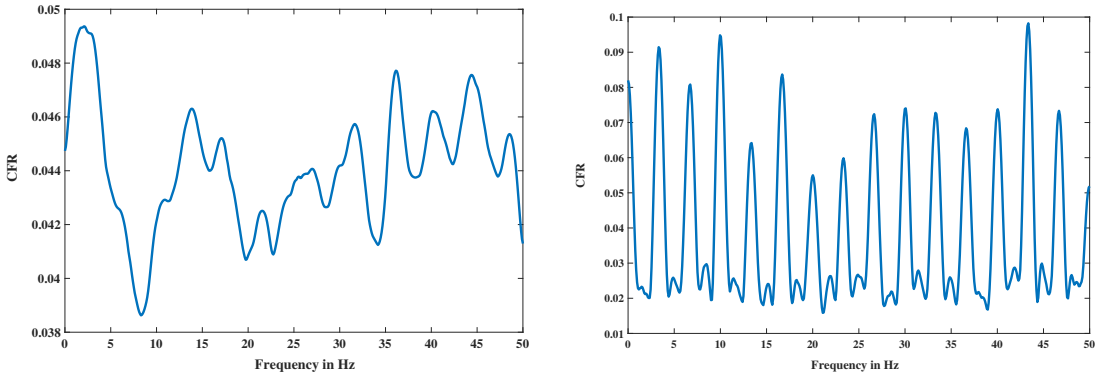


Figure 4.27: Cumulative frequency responses of analysis (*left*) and synthesis (*right*) filters.

Design concerns of the S2S network

Various configurations of the S2S network, type and nature of training targets and its effects on the performance are examined. With minimum number (zero) of hidden layers, S2S achieves better results than the baselines. All S2S configurations are performing

better and are computationally less expensive than the supervised baseline. The performance does not depend on the initialisation of parameters or filters and provided similar output after a fresh training process. This refers to the reliability of the S2S network, as the network training always converged within 50 epochs.

Linear correlation coefficient-based loss function provides the best performance. The results with other loss functions (mean squared error, correlation with a sigmoid activation) are also superior to the baselines. Being a synthesis-based approach, the sparse nature of the training target does not affect the performance. A small improvement in the results is obtained by learning a less-sparse Gaussian convoluted target. This also reduced the training time.

Multiple evaluation measures

Performance evaluation of a spike estimation algorithm should consider the overall shape of the spike information signal, monotonic relationship with the ground truth and, consider the accuracy of the time estimates of spikes with various thresholds. Multiple measures are considered because a single metric for spike inference is debatable as it only reveals limited parts of the spike inference (Kümmerer *et al.*, 2017).

Following the protocol used in spikefinder evaluation (Berens *et al.*, 2018), the Pearson correlation coefficient is considered as the primary evaluation measure as it considers the shape of spike density function into account. The computations are done at 25 Hz (40 ms bin width), enabling us to compare the performance with respect to the submissions of the contest. Correlation coefficient indicates how closely the spike information follows the discrete spike train. However, this measure is invariant under affine transformations. Thus it is impossible to interpret the outputs as spike counts or rates.

Spearman's rank correlation coefficient is considered as the secondary evaluation metric. It measures both the strength and the direction of association between two ranked variables. The ranking is based on the dynamic ranges of the spike information and discrete spike train. This correlation measure also considers the possible non-linear relationship between the variables. Spikefinder contest considered rank as the secondary evaluation measure for the training set. The rank values for the test set are presented for the top-performing algorithms in this work.

As the final evaluation measure, the area under receiver operating characteristics (AUROC or ROC) is used. This is the tertiary evaluation metric in the spikefinder contest. This measures the confidence with which output is classified as a spike or not. However, it is not sensitive to changes in the relative height of different parts of the spike information, and this measure alone is not adequate.

These evaluation measures collectively denote the similarity with the ground truth.

They are considered in the order of their preference as in the spikefinder challenge. The rank measure on the test set was not included in the challenge paper, although it was the secondary measure on the training sets. The information gain is not included as an evaluation measure as it is based on a model and is likely to be biased towards model-based approaches. It is also difficult to interpret information gain values (Kümmerer *et al.*, 2015).

Research findings and progress in spike inference

Despite having several different algorithmic strategies, all the approaches presented in the challenge are comparable in terms of results. Generative approaches with dataset-specific parameter tuning provided similar results to the more-flexible data-driven methods. All the supervised models in the challenge predict a single target value corresponding to spiking information. S2S network synthesises the spike information signal with the maximum similarity of power spectral density and is achieved by maximising the correlation coefficient. The major findings of this work are as follows:

1. S2S provides state-of-the-art spike estimation results: The proposed method has the best performance for both linear and non-linear correlation, although it has been trained only to maximise the Pearson correlation. The minimum variation between the train and test correlation is observed for S2S, although both the correlation values increased significantly compared to baselines. S2S network trained only with the GCaMP indicator performs comparable on the test data to the one trained with the entire spikefinder training set. The model does not overfit as indicated by the performance on unseen datasets.

2. S2S is a computationally efficient and reliable model: One of the major concerns in using a supervised approach for spike inference is its computational complexity. Generative approaches, on the other hand, required unique parametric settings for new datasets. For instance, this settings resulted in an average run time of 15 seconds for spike inference from single recording in the case of MLspike. All the best performing supervised algorithms of spikefinder had recurrent units in the architecture, which resulted in increased training time. The proposed S2S network was 50 times faster than the best performing supervised model when trained with GPU using Sun Grid Engine. The results do not depend on the initialisation of parameters or filters and provides reliable performance after a new training process.

3. Provides progress in spike inference field: Spikefinder contest discussed the improvement obtained in spike inference task by a community-based bench-marking compared to several individual efforts. The S2S network results in an improvement of 46% in primary evaluation measure which is two times that achieved

by the spikefinder contest (23%). This performance gain could help in further directions in spike estimation based on signal reconstruction strategies.

4.4.9 Comparison of spike estimation algorithms

We compare GDspike and S2S with other signal processing and machine learning algorithms for spike estimation. GDspike is a purely non-model based signal processing algorithm which is agnostic to the signal at hand. All other methods (Vogelstein, MLspike, STM and S2S) considered for comparison are model based approaches. GDspike performs better than Vogelstein and is comparable to MLspike on spike information signal (before making it discrete by thresholding). It performs inferior to MLspike while evaluating on the discrete spike train. However, MLspike is 13 times slower than both GDspike and Vogelstein, owing to the auto-calibration of parameters.

STM performs better than the signal-processing based methods, if no dataset specific parameter tweaking is performed for model parameters of MLspike. With such a parameter tuning, MLspike was the winner in the spikefinder contest, beating STM by 26% relative improvement in correlation. S2S outperforms all these approaches on spikefinder evaluations, with a relative improvement of 46% for the correlation measure. However, a dataset-specific training is needed for S2S to outperform signal processing methods if the dataset is purely unseen and not pre-processed to remove the linear trends.

4.4.10 Conclusion to spike estimation

We addressed the problem of spike estimation from neuronal signals using two methods: purely signal processing-based GDspike approach, and purely data-driven S2S approach. We establish that GD can be used for neuronal spike estimation if the bandwidth of the indicators fall within the range of bandwidth for which GD exhibits the HR property. This is true for all the existing indicators. Our neural network approach for spike estimation is inspired by attempts to end-to-end source separation. The proposed approach generates the sparse spike train at the output and has state-of-the-art performance for spike estimation task.

4.5 Summary

This chapter discussed ways to extract time-events from speech, audio and neuronal signals. Each of the TED tasks varies based on the temporal level of extraction, preprocessing and detection algorithm. However, all of them use the ability of GD to obtain sharp peak locations. This indicates that group delay is agnostic to the signal at hand

and could be applied for various segmentation tasks. MODGD-source is exploited for pitch estimation from speech signals. GD-based processing is used on the envelope of the music signal for percussive onset detection. Finally, this is exploited directly on the calcium fluorescence signal for spike estimation task. The proposed work benefit from relative slope changes of grating compression transform on variation in pitch. Tracking of pitch dynamics on modified-group delay gram is found to be useful for pitch estimation. An envelope extraction derived from AM-FM modelling is proposed based on the nature of the percussive onset and uses it along with GD processing for onset detection. The bandwidth of calcium fluorescence indicators lies within the allowable bandwidth at which high-resolution property can be observed. This observation resulted in a direct application of GD processing. The signal is made similar to the spike information by performing a simple-triangulation approximation. These novel applications shed light into the importance of phase-based feature representation for various segmentation tasks.

CHAPTER 5

Source Separation Systems

Signal extraction from a mixture of many signals considers each of the constituent components as source signals. This chapter presents source separation task and propose approaches containing a source separation stage in them. Source separation is a well-researched subject in audio signal processing. Various types of source separation are multi-speaker speech separation, speech enhancement, singing voice separation, instrument separation, etc. Several algorithms exist in the literature for each of these tasks. In this chapter, the general introduction to the source separation is discussed. The proposed methods are presented for musical source separation and separation-driven classification tasks.

This chapter is organised as follows: Section 5.1 provides an introduction to the task. Musical source separation is discussed in Section 5.2. A novel feature based on group delay is proposed for singing voice separation task and vocal-violin separation in Carnatic music is presented. This section also presents an end-to-end neural network for performing the singing voice separation task. Section 5.3 presents the first hybrid system based on the separation of percussion instrument from the musical mixture for onset detection from mixture segments, extending on the task discussed in Section 4.3. Automatic gender recognition (AGR) from noisy speech signals under weakly-supervised and language independent conditions is discussed in Section 5.4. A similar network is used for denoising in the first stage and a direct-waveform approach is explored for the classification of genders. Section 5.5 discusses the summary of the chapter.

5.1 Introduction

Source separation is generally associated with the "cocktail party problem", a scenario in which many speakers are speaking simultaneously in a cocktail party hall, and we wish to be attentive to one specific speaker. The ability of humans to perform this task flawlessly amid the presence of other sources is incredible and is of interest to the research community. Algorithms capable of extracting constituent sources from the mixture signal has been developed in various fields (Jung *et al.*, 2000; Cardoso, 1998; Lee, 1998). Mixing can be either linear or non-linear. Speech signal corrupted by noise can be considered as a mixture of the noise signal and the clean speech signal. Hence, any speech enhancement technique is also related to the source separation problem. The source separation problem can be posed as a single channel or multi-channel separation

task. In the single channel, only one audio signal track is available for the entire mixture, and the mixing ratio is generally unknown. On the other hand, multi-channel source separation deals with mixed signals from multiple channels. The audio in one of these channels also has traces from other sources. Estimating multiple sources from a single mixture is an ill-paused problem as there are more sources than the mixture(s). In monaural source separation, both the mixing coefficient and the source-specific details need to be extracted from a single mixed signal. Thus, it is a harder problem than multi-channel separation. This chapter aims at the single channel source separation by considering the linear mixtures. Information about mixing sources or the mixing process is usually not known in a source separation scenario. Hence, such a problem is known as blind source separation (BSS).

5.1.1 Overview

The following four fundamental methods can summarise the basic principles of various source separation algorithms.

- A popular signal processing method relies on statistical independence, sparseness or non-Gaussianity of the signals. When the source signals are assumed to be without a temporal structure and are statistically independent, the statistics at higher-orders are used for solving the source separation task. In such a case, the method does not allow more than one Gaussian source.

- The second approach exploits both non-stationarity properties and second-order statistics (SOS). These methods allow the separation of sources with a Gaussian distribution, though with identical power spectra shapes. However, they are limited by sources which are having an identical non-stationarity properties.

- Less restrictive conditions than statistical independence can be used for BSS if the sources have temporal structures. For estimating the mixing matrix and the individual sources, the Second-order statistics (SOS) are often sufficient. These methods do not allow the separation of sources with identical power spectra shapes.

- The fourth method makes use of the various diversities of signals such as time, frequency, or a mix of both. This class of algorithms are known as time-frequency component analyser (TFCA). It decomposes the signal into specific time-frequency (TF) components and computes the corresponding representations of the source signals. The source signals are assumed to be well-structured and sparse localised signals in the TF representation (Belouchrani and Amin, 1996). This method helps in improving the source estimates by suppressing the artefacts and minimising the interference using TF masking. This results in filtering out of the unwanted components. Recent methods perform this TF mask estimation using machine learning techniques.

Since all the above methods are mathematical tools, incorporation of *a priori* knowledge and processing steps before and after the method implementations become important. Typical pre-processing techniques aim to separate out components which differ in their feature space representations. Fourier transform based techniques, signal sparsification, Factor Analysis and Principal Component Analysis are used in this step. In the post-processing step, the original signals are reconstructed. Most of the above mentioned approaches are a part of unsupervised source separation techniques.

5.1.1.1 BSS using Time-Frequency Representations

Among source separation tasks, this chapter specifically focuses on time-frequency representation (TFR) based source separation (Belouchrani and Amin, 1996). The best performance achievable where the reference signals are known is to find the best de-mixing filters or best time-frequency masks for a given mixture signal. Even with the best de-mixing filters, the performance is limited. The TFR could be learned using machine learning approaches or obtained using unsupervised methods.

5.1.1.2 Filtering techniques

Source separation by harmonic reconstruction from mixed signal employ Computational Auditory Scene Analysis (CASA) cues such as harmonicity in TFR, common onset/offset times etc. Unsupervised clustering algorithms can be used for building average harmonic structure (Duan *et al.*, 2008) of the sources and corresponding sources are extracted (Weiss and Ellis, 2006) using them.

One shortcoming of pitch based harmonic reconstruction is that it can hardly separate sources playing the same pitch or with many overlapping partials. To address this problem, a T-F smoothness constraint is added on the source components in (Virtanen, 2003), while spectral filtering techniques are used to allocate energy for overlapping partials in (Every and Szymanski, 2006). However, they both require knowledge of the pitch values of the sources (Bay and Beauchamp, 2007). Hence, multi-pitch estimation acts as another bottleneck in BSS.

Filtering methods can be broadly classified as:

(a) Beamforming: It involves filtering the mixture channels by stationary filters and summing them together. Optimum de-mixing filters could be found out from the reference signals. It works best for the static or slightly convolutive mixture.

(b) Time-Frequency Masking: It involves computing the STFTs of the mixture channels, multiplying them by time-frequency masks containing gains between 0 and 1 and inverting the resulting STFTs. The most popular masking rules are adaptive Wiener filtering and binary masking. This generates more artefacts than beam forming(Bay and Beauchamp, 2007).

5.1.1.3 Deep learning methods

Among the various supervised and model-based approaches to source separation, the latest attention has been towards the deep neural network (DNN) models which learn the time-frequency representations of individual sources. DNN has been applied recently to BSS problems with different model architectures (Boulanger-Lewandowski *et al.*, 2014; Huang *et al.*, 2014*a*) where the models learn the mapping between the mixed signal and the separated signals. Huang et al. proposed Deep Recurrent Neural Network (DRNN) for monaural Blind Source Separation (BSS) (Huang *et al.*, 2014*a*) in which both the sources are simultaneously modelled. Time-frequency masking is employed to make the sum of the prediction results equal to that of the original mixture. In (Weninger *et al.*, 2014), long-short-term memory (LSTM)-based DRNNs are introduced for source separation of speech signals. The soft T-F mask (Huang *et al.*, 2012) is applied to the magnitude spectrum of the mixture signal to obtain the separated spectra. This masking function is added as an additional deterministic layer, and the network is jointly optimised with the masking function.

As a learning approach, DNNs do not require any task-specific assumptions and prior source knowledge which may not always be true in the real world applications. The network parameters are directly learned from the data. For many of the audio applications, state-of-the-art results are obtained using deep learning (Hinton *et al.*, 2012; Glorot *et al.*, 2011). DNN has been applied recently to BSS problems with different model architectures (Boulanger-Lewandowski *et al.*, 2014; Huang *et al.*, 2014*a*) where the models learn the mapping between the mixed signal and the separated signals.

5.1.2 Sequence-to-sequence Models

Sequence-to-sequence neural network models are developed recently for classification and regression tasks. The feature extraction step, which is typically performed as a pre-processing step for data-driven methods, is removed in a sequence-to-sequence model. These approaches model the sequences in an end-to-end fashion. In audio and natural language processing, such models have commendable performance. Sequence-to-sequence models are used for text summarising using RNNs in (Nallapati *et al.*, 2016). This model takes a paragraph of text as input and outputs its summary.

The sequence-to-sequence models are developed recently for audio signal processing. This includes speech recognition, speaker recognition and verification, and speech source separation. The sequence-to-sequence models consider the sequence information at the output. For automatic speech recognition, these models predict a sequence of symbols for a given acoustic signal (Sutskever *et al.*, 2014; Chan *et al.*, 2016; Chiu *et al.*, 2018). For example, Listen Attend and Spell (LAS) model (Chan *et al.*, 2016) use attention mechanism along-with the sequence modelling for improved estimates of

the character sequence.

For speech separation, the speech of the target speaker is predicted from the overlapped speech mixture directly using the waveform for feature extraction and signal reconstruction (Venkataramani *et al.*, 2017; Luo and Mesgarani, 2018). Clean speech is directly learned in this manner for speech enhancement tasks (Fu *et al.*, 2017; Rethage *et al.*, 2018). These models work on the raw waveform, hence annihilate the need of hand engineered features. These models also provide the flexibility to choose task-specific objective functions for training which implicitly consider the time-varying signal. It is trivial to analyse the filters to understand the learning trends in the temporal and frequency domains. Inspired by these approaches, this chapter proposes a new end-to-end framework for musical source separation in Section 5.2.5.

5.1.3 Recurrent neural network for BSS

Recurrent neural networks (RNN) are characterised by temporal connections between the layers of two neural networks. These are used to capture the contextual information among the sequential data. However, the hierarchical processing is limited owing to the system lacking hidden layers. Deep RNNs (DRNNs) provide this information at multiple time scales. Figure 5.1 shows a typical deep recurrent neural network architecture used in BSS (Huang *et al.*, 2014*b*). l-DRNN is the one with a temporal connection at l^{th} layer. The temporal connection is present at every layer of the stacked DRNN. For an l-DRNN, the hidden activation at level l and time t is given by:

$$h_t^l = f_h(x_t, h_{t-1}^l) \tag{5.1}$$

$$= \phi_l(U_l h_{t-1}^l) + W^l \phi_{l-1}(W^{l-1}(...\phi_1(W^1 x_t))), \tag{5.2}$$

The output value y_t is then obtained as,

$$y_t = f_0(h_t^l) \tag{5.3}$$

$$= W^L \phi_{L-1}(W^{L-1}(...\phi_l(W^l h_t^l))), \tag{5.4}$$

where x_t is the input to the network at time t, W^l is the weight matrix for the l^{th} layer, U^l is the weight matrix for the recurrent connection at the l^{th} layer and $\phi_l(\cdot)$ is the nonlinear activation function. Huang et al. (Huang *et al.*, 2014*b*) empirically found that the rectified linear unit (ReLU: $f(x) = max(0, x)$) performs better compared to using a sigmoid$(\cdot)$ or tanh$(\cdot)$ activation function.

Feature vector x_t is given as the input to the network to obtain the source estimates, $y_t'^1$ and $y_t'^2$. The network parameters are optimised by minimising the Mean Squared Error (MSE) objective function and Kullback Leibler divergence (KL). This discriminative objective function not only increases the similarity between the prediction and

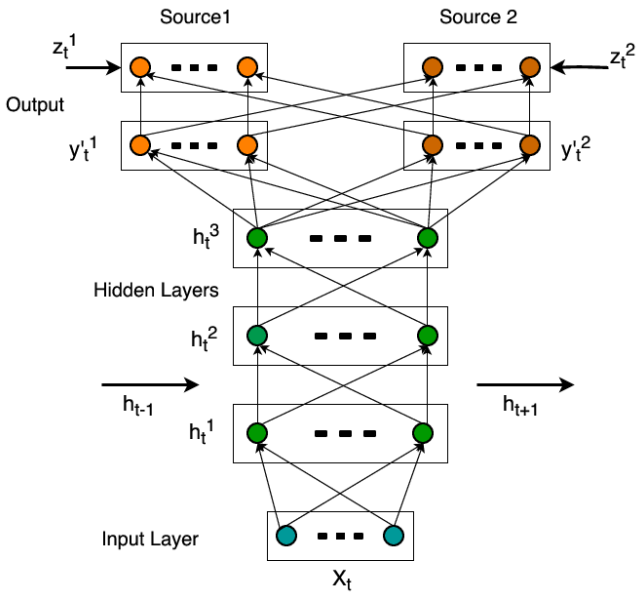


Figure 5.1: DRNN used for source separation (Redrawn from (Huang *et al.*, 2014*b*)).

target but also decreases the similarity between the prediction and the targets of other sources.

The objective function is given by:

$$||\widehat{y}_{1t} - y_{1t}||_2^2 - \gamma||\widehat{y}_{1t} - y_{2t}||_2^2 + ||\widehat{y}_{2t} - y_{2t}||_2^2 - \gamma||\widehat{y}_{2t} - y_{1t}||_2^2 \tag{5.5}$$

and the divergence criterion used is:

$$D(y_{1t}||\widehat{y}_{1t}) - \gamma D(y_{1t}||\widehat{y}_{2t}) + D(y_{2t}||\widehat{y}_{2t}) - \gamma D(y_{2t}||\widehat{y}_{1t}), \tag{5.6}$$

where $D(A||B)$ is the KL divergence between A and B. The γ parameter is chosen based on development data performance. This chapter utilise the recurrent neural network architecture for showing the benefit of GD feature for musical source separation task in Section 5.2.1.

5.1.4 Challenges

Source separation is an ill-posed problem which requires assumptions on the mixing matrix based on the modelling (linear or nonlinear) and prior information about the sources. It includes statistical independence assumption in Independent Component Analysis (ICA), non-negativity constraint in Non-Negative Matrix Factorisation (NMF), and sources colouration or non-stationarity in joint diagonalisation. In one of the unsupervised approaches, source separation is achieved using multi-pitch estimation. This itself is a more laborious task because of the presence of multiple speakers in the mixture signal. A proper multi-pitch estimation algorithm is needed as the input

for the unsupervised task which, in turn, is a challenging problem.

Timbre characteristics are different for different sounds. Appropriate timbre models are built specifically to each speaker in a supervised approach to speech separation. In unsupervised approach, pitch harmonics are used for reconstructing the individual estimates because significant energy is contained in the fundamental and its harmonic frequencies. Since more ”activity” of the sound is around pitch harmonics, reconstruction algorithms are used for estimating the individual signals. But it is limited by the pitch overlap, identical harmonic frequencies and unknown filter parameters. Also, since the phase spectrum is not additive, it is impossible to get the individual phases for reconstruction.

5.1.5 Applications

A wide range of applications of source separation includes but not limited to signal extraction, enhancement, denoising, model reduction and classification problems. Source separation is widely used as a stand-alone as well as pre-processing stage in MIR applications. It is used for the separation of singing voice, instrument and karaoke from the musical mixture and also as a pre-processing stage for higher level retrieval tasks such as finding the rhythmic and melodic patterns of the mixtures.

Source separation is essential in speech technology research. It is used as a pre-processing stage for noise-robust ASR systems wherein separation of background noise is performed using a separate denoising/speech enhancement framework. This is also used as a stand-alone technique for multi-speaker speech separation task. This is applied to speaker-dependent personal devices which work under multi-speaker conditions and noisy environments, speech diarization where the constituents of an overlapped speech are detected and, enhancing the separation performance of hearing aids.

5.1.6 Features used for BSS

These networks are modelled to learn the time-frequency patterns for each of the sources from the raw mixture signal. Separability of these patterns in the feature domain enhances the source separation quality. At present, magnitude spectrum based features such as Mel Frequency Cepstral Coefficients (MFCC), logMel (Huang *et al.*, 2014*a*,*b*) and the magnitude spectrum itself (Mysore *et al.*, 2010; Simpson, 2015; Huang *et al.*, 2014*b*) are used to learn the optimum time-frequency mask. In (Huang *et al.*, 2014*a*), MFCC features that are commonly used for other audio applications are employed, while in (Huang *et al.*, 2015), logMel features are used owing to the success of logMel features in Automatic Speech Recognition (ASR) (Li *et al.*, 2012). However, the performance was better for the magnitude spectrum feature compared to MFCC and logMel features (Huang *et al.*, 2014*b*).

For music source separation, spectrum as a feature has yielded the most promising results. Performance gets degraded when the individual pitch trajectories overlap, or the formants of the different sources are closer. Phase spectrum based group delay function has been successfully used in Music Information Retrieval (MIR) tasks such as tonic identification (Bellur and Murthy, 2013*b*), musical onset detection (Kumar *et al.*, 2015) and melody mono pitch extraction (Rajan and Murthy, 2013).

5.1.7 Performance measures

There are subjective and objective measures for the evaluation of source separation performance. Quantitative BSS evaluation metrics are presented in (Vincent *et al.*, 2006). Depending on the exact application, different distortions can be allowed between an estimated source and the true source signals. It consists of determining the contribution of the other source (interference) and the change in the original source (artefacts) in the estimated source.

The estimated source (s_j) is decomposed into orthogonal sub-space projections which result in three projectors, P_{s_j}, P_S and $P_{S,n}$ corresponding to the desired source, total source space and the noisy overall space, respectively. The source signal s_j is then decomposed as:

$$S_{target} := P_{s_j}\hat{s}_j \tag{5.7}$$

$$e_{interf} := P_S\hat{s}_j - P_{s_j}\hat{s}_j \tag{5.8}$$

$$e_{noise} := P_{S,n}\hat{s}_j - P_S\hat{s}_j \tag{5.9}$$

$$e_{artif} := P_{S,n}\hat{s}_j - P_S\hat{s}_j \tag{5.10}$$

These projections are calculated based on the assumption that the noise signals are mutually orthogonal and orthogonal to each source as well. From above equations, energy ratios (in decibels) are computed as follows,

1. Signal-to-Distortion Ratio (SDR)
The measure of the overall performance of the estimated source.

$$SDR := 10log_{10}\frac{||s_{target}||^2}{||e_{interf} + e_{noise} + e_{artif}||^2} \tag{5.11}$$

2. Signal-to-Interference Ratio (SIR)
It is the ratio of the true source to the interference of the other sources.

$$SIR := 10log_{10}\frac{||s_{target}||^2}{||e_{interf}||^2} \tag{5.12}$$

3. Signal-to-Artifact Ratio (SAR)

It is the measure of the artefacts introduced by the method.

$$SIR := 10log_{10}\frac{||s_{target} + |e_{interf} + e_{noise}||^2}{||e_{artif}||^2} \tag{5.13}$$

These measures are inspired by the definition of the signal to noise ratio (SNR) with some modifications.

Normalised SDR (NSDR) is defined by (Huang *et al.*, 2014*b*) as:

$$NSDR(\widehat{v}, v, x) = SDR(\widehat{v}, v) - SDR(x, v), \tag{5.14}$$

where x is the mixture, $\widehat{v}$ and v are the estimated source and the actual clean source respectively. Improvement of the SDR between the mixture and the separated source is reflected in NSDR. The performance measures are weighted by the length of the audio and averaged across the entire test set to obtain the corresponding global measures such as Global SIR (GSIR), Global SAR (GSAR), and Global NSDR (GNSDR).

The subjective or qualitative analysis includes the following measures:

1. **Perceptual Evaluation of Speech Quality (PESQ)**: This measure uses several levels of analysis in an attempt to mimic the human perception.
2. **Other measures:** Several other measures such as weighted spectral slope (WSS), total relative distortion (TRD), cepstral distortion (CD), non-linear PESQ, Short-term Objective Intelligibility (STOI) and a combination of these are also used for evaluation.

5.2 Musical Source Separation

This section proposes two approaches for musical source separation tasks. We mainly focus on the phase-based Modified Group Delay (MODGD) feature (Yegnanarayana *et al.*, 1991) for learning the time-frequency mask in BSS as opposed to conventional magnitude spectrum based features. Further, this section propose to use a signal-to-signal conversion neural network for singing voice separation task.

Features based on MODGD function have been used for speaker verification and it is observed in (Asha *et al.*, 2014) that MODGD is the preferred feature to MFCC for a large number of speakers. Clearly, the timbre of the speaker is captured by this feature. The sources correspond to different timbres in the source separation problem. The MODGDgram feature is proposed which is obtained by concatenating MODGD function over the consecutive frames in DRNN architecture (Huang *et al.*, 2014*b*) and discuss the performance and the computational/architectural advantages over the spectrum feature.

Algorithms introduced in the literature still use mask-based separation for the singing

voice. Instrument separation on waveform domain is attempted by (Lluís *et al.*, 2018) in which a wavenet model initially proposed for speech enhancement (Rethage *et al.*, 2018) is adapted for estimating the individual instruments from a mixed musical segment. Hence, it consists of non-causal dilated convolutions, and it generates the instrument channels, not with a phase-mixing procedure. The performance is compared with a fully-convolutional neural network (Chandna *et al.*, 2017), deepConvSep, and reported improvements over spectrum based instrument separation approaches. S2S network is proposed for singing voice separation. The benefit of this network is discussed in comparison to feature-based singing voice separation schemes.

5.2.1 BSS with MODGDgram

The architecture of DRNN shown in Figure 5.1 is used with the MODGD feature for music source separation. The input feature to the DRNN network is the modified group delay-gram (MODGDgram) which is obtained by concatenating MODGD function of the successive frames. The time-frequency mask learned from them are used to filter the mixture magnitude spectrum to obtain the individual source spectra. The MODGD is computed from the signal and its time-weighted version, as given in Equation 2.19 and Equation 2.21. The moving average smoothing function is used in place of cepstral smoothing function (Murthy and Yegnanarayana, 2011) as the former is more robust to zeros in the frequency domain. As regions around the formants are important for timbre, the powers for the positive peaks (α_1) are set different from that of the negative peaks (α_2).

Figure 5.2 compares the spectrogram and the MODGDgram of the sources and their linear mixtures used in singing voice separation for a music segment from the MIR-1K dataset. The first row represents the spectrogram and the second row represents log-MODGDgram. The third column is the linear mixture of the first (singing voice) and second (background music) columns. The time-frames are squeezed to make the pitch trajectories visible. FFT size is chosen to be 512, and the lower 100 bins are used for plotting since it has most of the melodic information. It should be noted that the mixture MODGDgram preserves the harmonics of the sources better than the mixture spectrogram. Observe from the figure that the dynamic range is higher for the MODGDgram compared to that of the spectrum, in that pitch trajectory stands out with respect to the background. The MODGD feature has a comparable computational complexity to that of the spectrum for the same input dimension.

5.2.2 Experiments

The source separation performance is evaluated using the MODGD feature on two music source separation tasks: singing voice separation and vocal-violin separation. A

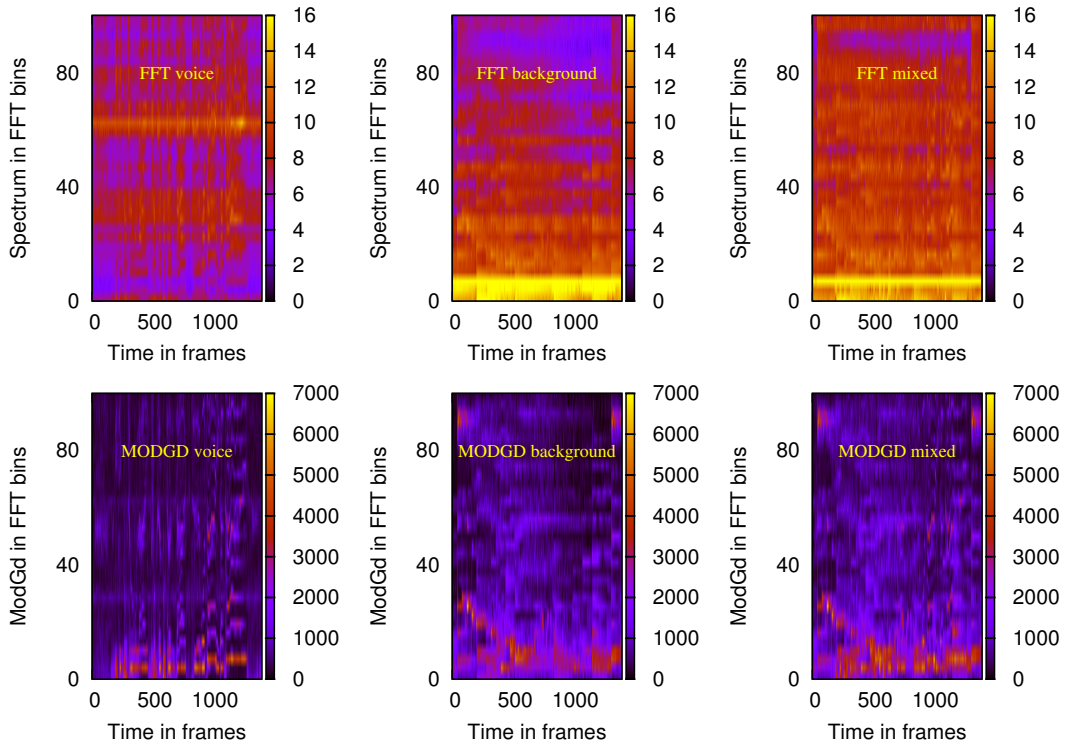


Figure 5.2: Feature representations of the clip Ani_1_01.wav from MIR-1K dataset.

three layer DRNN architecture with discriminative objective function (Equation 5.5) is used in the experiments. The maximum epoch is set to 400 in each configuration.

Evaluation Metrics

The source separation quality is measured using three quantitative measures based on BSS-EVAL 3.0 metrics (Vincent *et al.*, 2006). Normalised SDR (NSDR) is defined by (Huang *et al.*, 2014*b*) as:

$$NSDR(\widehat{v}, v, x) = SDR(\widehat{v}, v) - SDR(x, v), \tag{5.15}$$

where x is the mixture, $\widehat{v}$ and v are the estimated source and the actual clean source respectively. Improvement of the SDR between the mixture and the separated source is reflected in NSDR. The Test clips are weighted by their length and their weighted means are used to represent the overall performance via Global measures (GSAR, GSIR and GNSDR).

Datasets used

For the singing voice separation task, the MIR-1K dataset (Hsu and Jang, 2010) is used to evaluate the performance of the MODGD feature. It consists of thousand song clips at 16 kHz sampling rate with a duration of 4 to 13 seconds. Each clip contains the singing voice and the background music in different channels. These clips were

extracted from 110 Chinese karaoke songs performed by male and female amateurs. Training set consists of 171 clips sung by one male and one female (”abjones” and ”amy”). The development set contains 4 clips sung by the same singers, following the same framework as in (Huang *et al.*, 2014*b*). The test set consists of the remaining 825 clips from 17 amateurs. Channels are mixed at 0 dB SNR and our aim is to separate the singing voice from the background music.

Since there was no dataset specifically for Carnatic music source separation, one dataset is created for vocal-violin separation task using publicly available MusicBrainz corpus [1] released as a part of CompMusic project [2]. From a concert of 2 hours and 3 minutes duration, 77 musical clips are extracted with the duration ranging from 2 to 23 seconds. The recorded data is a two channel signal with the vocal in one channel and the lead instrument (violin) in the other. These are mixed at equal energy levels to obtain a single channel mixture signal. The training data consists of randomly selected 54 clips, the development set contains 3 clips and the test set consists of remaining 20 clips.

5.2.3 Singing voice separation in MIR-1K dataset

Experiments are performed with both the MODGDgram and magnitude spectrogram features. The spectral representation is extracted using 1024 point short time Fourier transform (STFT) with an overlap of 50%. Following (Huang *et al.*, 2014*b*), a $32ms$ window with $16ms$ frame shift is used for calculating the features. This longer context and shift are observed to be beneficial for musical source separation task as the T-F structure is more prominent in comparison to speech signals. Since the context features can further improve the performance, A contextual window of 3 frames is used. In the modified group delay computation, smoothing parameter is set to 5, and the group delay scales (α_i) are set to 1.2 and 0.45, as obtained from the multi-pitch task (Rajan and Murthy, 2016).

The performance of the MODGD feature is compared with that of the magnitude spectrum feature on several aspects. In terms of complexity (Table 5.1), it is observed that the architecture with just 500 hidden nodes per layer performs similarly to that of the architecture with 1000 nodes per layer with the spectrum feature. Hence, a network with 1500 fewer hidden nodes is sufficient to achieve the same performance, i.e., training and testing times are halved.

The best results (2-DRNN) obtained using the spectrum feature (Huang *et al.*, 2014*b*) is also compared with our approach in Table 5.1. For the same setting, MODGDgram feature gives similar results for SAR and SDR and shows a relative improvement of 4.9%dB for SIR over magnitude spectrum. This is because the mask is learned from the

[1] https://musicbrainz.org/

[2] https://compmusic.upf.edu/

Table 5.1: Performance measures with 2-DRNN.

Feature	Hidden units per layer	GNSDR	GSIR	GSAR
MODGD	500	**7.15**	**13.46**	**9.11**
Spectrum	500	5.74	12.15	7.62
MODGD	1000	**7.50**	**13.73**	9.45
Spectrum	1000	7.45	13.08	9.68

group delay domain, where the resolution is higher than the spectrum. Note that there is not much improvement from 500 to 1000 hidden units per layer, which suggests intelligent separation is possible with a simpler network with MODGDgram feature.

Table 5.2: Results with various configurations of DRNN.

Architecture	Feature	GNSDR	GSIR	GSAR
1-DRNN	Spectrum	7.21	12.76	**9.56**
	MODGD	**7.26**	**12.93**	9.42
2-DRNN	Spectrum	7.45	13.08	**9.68**
	MODGD	**7.50**	**13.73**	9.45
3-DRNN	Spectrum	**7.09**	11.69	**10.00**
	MODGD	6.92	**12.27**	9.26
stacked DRNN	Spectrum	7.15	12.79	**9.39**
	MODGD	**7.31**	**13.45**	9.30

Table 5.2 shows the performance of the feature on several RNN configurations compared to the spectrum. Better SIR ratio is achieved for *all* the configurations with similar values for other measures. Thus, MODGDgram improves the quality of separation irrespective of the model configurations.

5.2.4 Vocal-Violin separation in Carnatic music dataset

In a concert, the vocal and all the accompanying instruments are tuned to the same base frequency called *tonic* frequency. This can lead to overlapping of the pitch frequencies corresponding to vocal and other instruments. Hence, Carnatic music source separation is not possible with simple dictionary learning methods. This is the first attempt at source separation for a live Carnatic music concert with no constraint on the data.

The results obtained with MODGDgram and spectrogram features are compared on an architecture with 1000 hidden units per layer. The architecture of DRNN with a temporal connection at 1^{st} hidden layer (1-DRNN) is used to obtain the results. Other experimental settings are made similar to that of singing voice separation task. From Table 5.3, it is observed that the performance of both the features are almost equal, with

MODGDgram feature giving slightly better GSIR. This is also reflected in the GNSDR.

Table 5.3: DRNN performance in the Carnatic music dataset.

Feature	GNSDR	GSIR	GSAR
MODGD	**9.42**	**13.72**	11.76
Spectrum	9.38	13.55	**11.80**

From the experiments it can be inferred that the MODGDgram can replace the spectrogram feature for the music source separation task in the state-of-the-art DRNN architecture because of two major reasons: First, it gives better GSIR values and second, the MODGDgram based DRNN is less complex, resulting in a reduction of the computation time by 50% in the best configuration of the architecture. This work also conjectures that the higher resolution property helps in learning the average time-frequency trajectories with a simpler network.

5.2.5 Singing Voice Separation using S2S

A singing voice separation system based on signal-to-signal conversion is proposed. On the contrary to the instrument separation, singing voice estimation focuses on the clean vocal estimates. The proposed system is different from the approach by (Lluís *et al.*, 2018) in several aspects such as the use of masking function, the hybrid structure consisting of time-distributed hidden layers and the use of explicit phase representation. Such a network has not been developed for singing voice separation so far[3]. This end-to-end network has an encoder-decoder structure with a set of hidden layers sandwiched between them. The filters in the analysis and synthesis layers together learn the T-F characteristics specific to the singing voice.

Adapting S2S for singing voice separation

The S2S architecture used for spike estimation in Section 4.4 is adapted to perform singing voice separation. This comprises a smoothing operation after the convolutional layer, which generates a smoother time-frequency (T-F) structure and thereby annihilating the phase variations from passing through the mask estimating hidden layers. The output of the smoothing function has a ReLU non-linearity making it a magnitude-based representation (M), based on the bases learned from the mixed signal.

This magnitude representation is fed to the time-distributed hidden layers which operate on all the temporal frames in the input window at once. This avoids the process-

[3] At the time of writing this thesis

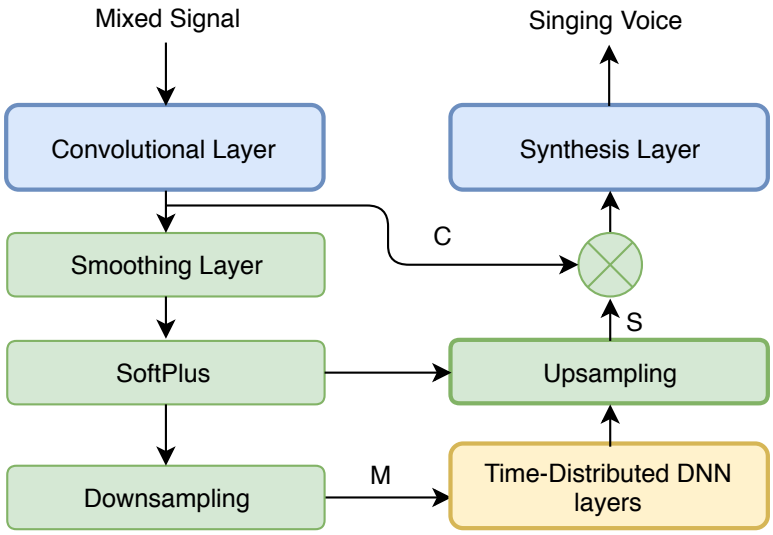


Figure 5.3: S2S adapted for singing voice separation.

ing of each of the input windows separately. The network generates signals of duration equal to the length of the input signal. The magnitude-based representation (M) divides the original convolutional output (C) to obtain a phase-based representation. The output of the hidden layers is then mixed with this representation in an element-wise fashion to generate a real transformation of the estimated signal. In the final stage, a single output is generated for musical source separation by a de-convolution operation as described before. The architecture of the adapted S2S is shown in Figure 5.3.

Evaluations

Similar to the evaluations performed for MODGDgram based singing voice separation, the experiments are performed on the publicly available singing voice separation dataset; MIR-1K (Hsu and Jang, 2010). Other experimental details are kept as the same as in RNN based separation system. The S2S architecture use 1600 samples (corresponding to 100 ms) as the width of the convolutional layer filter. Hop size is 160 samples (10 ms). There are 1024 filters in both convolutional and synthesis layers. Three hidden layers with 512 points per layer are employed in time-distributed fashion. Hence down-sampling and up-sampling processes are introduced in the architecture to map from 1024 filter outputs to 512 units and vice-versa.

Evaluation procedure

The S2S network is trained until the error on the validation set reaches 5e-4 or the number of epochs reaches 1000K iterations, whichever is the first one to achieve. For each epoch, a mini batch size of 16 is used. Each training example consists of randomly selected 2 sec of mixed signal and its corresponding singing voice channel taken from the training corpus. The number of batches per epoch is kept as 55 such that it cov-

ers almost the entire dataset in a single epoch. The network is optimised to minimise the loss function which is based on the signal similarities with the ground truth voice. S2S uses direct signal similarity measures for improved separation performance. A correlation based objective function is used so as to maximise the similarity between the estimated and original signing voice. This is directly proportional to SDR measure (Venkataramani *et al.*, 2017), which is evaluated in the test phase.

Systems considered

A traditional STFT based singing voice separation system based on DNN is trained with a single source and no contextual window to serve as the baseline of the proposed approach. The best-possible results on the DNN with data augmentation, contextual window and multiple source estimates is used as the second baseline for the test set. The test results for the second baseline are taken from (Huang *et al.*, 2014*b*).

There are two variants in the proposed system. System 1 consists of S2S in which there is no constrain on the synthesis and analysis filters (S2S-1). System 2 has the constraint that, the analysis operation is the reverse of the synthesis operation (Venkataramani *et al.*, 2017). This is employed with the hope of better initialisation and faster learning (S2S-2).

Results

Table 5.4 shows the performance of development and test datasets for the baselines and the proposed systems. Singing voice separation achieves the state-of-the-art results using end-to-end S2S. The proposed methods perform better than the DNN baselines. DNN-2 uses the best possible DNN configuration with data augmentation, discriminative training and joint optimisation of masks. Proposed methods still perform better than DNN-2 in spite of not having data augmentation or contextual window. This is because of the direct feature learning aimed at maximizing the signal-based similarity measures, in comparison to spectral masks which only looks at magnitude spectrum of the estimated source.

Table 5.4: Performance measures with proposed approach.

System	Dev Set			Test Set		
	GNSDR	GSIR	GSAR	GNSDR	GSIR	GSAR
Baseline: DNN-1	5.60	21.40	6.20	5.00	15.87	5.99
Baseline: DNN-2	-	-	-	7.45	13.08	9.68
Proposed: S2S-1	9.20	24.98	9.57	7.65	22.21	8.18
Proposed: S2S-2	9.62	24.75	9.97	7.72	22.71	8.22

Global SDR is calculated for both development and test sets. The proposed S2S has a 2.65 dB absolute improvement over the spectrum-based singing voice separation system (DNN-1). Corresponding improvement is observed for the development data as well. S2S performs better than the mask-based approach (DNN-1) for all of the performance measures, even with a similar intermediate layer architecture. This validates the claim that S2S learns source-specific bases depending on the training data, on the contrary to mask based approaches where the bases are fixed DFT dimensions.

DNN-2 achieves better performance than DNN-1 owing to data augmentation and contextual windowing. However, the performance is still inferior to end-to-end source separation architecture. S2S has a notable performance for GSIR in particular. This is reasonable as the contributions from other source is almost nullified by using signal-based similarity measures. In mask-based approach, the phases of input sources are still mixed at the output. This leads to reduced GSIR value in comparison to S2S.

S2S-2 performs better than S2S-1 as the forward transform is made to be the inverse of the reverse transform, though the weights of the bases are learned from the dataset. This implicitly makes a better initialization of reverse transform bases. This constraint helped in achieving a faster convergence. This shows the potential of S2S for singing voice separation and possibly a larger class of audio source separation systems.

5.3 Hybrid Systems: 1. Percussive Onset Detection

Hybrid approaches are presented for classification tasks which are inspired by the separation framework used in Section 5.2. Hybrid systems consist of more than one stage for performing the tasks. The hybrid structures discussed here consist of two parts; a separation stage and then a classification stage. Hybrid systems also mean that we can generalize the tasks into these two stages. Two-hybrid systems are considered in which source separation framework are used as the pre-processing stage. This section considers percussive onset detection from the musical mixture. Complex rhythmic patterns associated with Carnatic music are revealed from the stroke locations of percussion instruments. However, a comprehensive approach to the detection of these locations from composition items is lacking. This is a challenging problem since the melodic sounds (typically vocal and violin) generate soft-onset locations which result in a number of false alarms.

In this task, a separation-driven onset detection approach is proposed. Percussive separation is performed using a DRNN in the first stage. A single model is used to separate the percussive vs the non-percussive sounds using discriminative training and time-frequency masking. This is then followed by an onset detection stage based on group delay processing on the separated percussive track. The proposed approach is evaluated on a large dataset of live Carnatic music concert recordings and compared against

percussive separation and onset detection baselines. The separation performance is significantly better than that of Harmonic-Percussive Separation (HPS) algorithm, and onset detection performance is better than the state-of-the-art convolutional neural network based algorithm. The proposed approach has an absolute improvement of 18.4% compared with the detection algorithm applied directly on the composition items.

5.3.1 Introduction

Detecting and characterising musical events is an important task in Music Information Retrieval (MIR), especially in Carnatic music, which has a rich rhythm repertoire. There are seven different types of repeating rhythmic patterns known as *tāla*s, which when combined with 5 *jāti*s give rise to 35 combinations of rhythmic cycles of fixed intervals. By incorporating 5 further variations called *gati*/*nadai*, 175 rhythmic cycles are obtained (Humble, 2002). A *tāla* cycle is made up of *mātrā*s, which in turn are made up of *akṣharā*s or strokes at the fundamental level. Another complexity in Carnatic music is that the start of the *tāla* cycle and of the composition need not be synchronous. Nevertheless, percussion keeps track of *rhythm*. The detection of percussive syllable locations aids higher level retrieval tasks such as *akṣharā* transcription, *sama* (start of *tāla*) and *eḍuppu* (start of composition) detection and *tāla* tracking.

Three sections characterise every item in a Carnatic music concert. Every item has, at its core, a composition. A lyrical composition section is usually performed by the lead performer, accompanying violinist and the percussion artist altogether. This section is optionally preceded by a pure melody section (*ālāpana*) in which only the lead performer and the accompanying violinist perform. The composition section is optionally followed by a pure percussion section (*tani āvarthanam*). Onset detection and *akṣharā* transcription in *tani āvarthanam*s are performed in (Kumar *et al.*, 2015), and (Kuriakose *et al.*, 2015) respectively. Percussive onset detection for an entire concert that is made up of 10-12 items, each associated with its own *tāla* cycle, is still challenging as the composition items are made up of ensembles of a lead vocal, violin/ensembles of the lead instrument(s) and percussion.

Onset detection in polyphonic music/ensemble of percussion either use audio features directly (Benetos and Dixon, 2011) or performs detection on the separated sources. Dictionary learning-based methods using templates are employed in the separation stage in certain music traditions (Tian *et al.*, 2014; Goto and Muraoka, 1994). Harmonic/percussive separation (HPS) from the audio mixture is successfully attempted on Western music in (Fitzgerald, 2010) and (Fitzgerald *et al.*, 2014). Onset detection of notes is performed on polyphonic music in (Benetos and Dixon, 2011) for transcription. Efficient percussive onset detection on monaural music mixtures is still a challenging problem. The current approaches lead to a significant number of false positives, owing to the difficulty in detecting only the percussive syllables with varying amplitudes and

the presence of melodic voices.

In a Carnatic music concert, the lead artist and all the accompanying instruments are tuned to the same base frequency called *tonic* frequency and it may vary for each concert. This leads to the overlapping of pitch trajectories. The bases do not vary over time in the case of dictionary-based separation methods, leading to a limited performance in Carnatic music renderings. HPS model (Fitzgerald, 2010) does not account for the melodic component and variation of *tonic* across the concerts. The state-of-the-art solo onset detection techniques, when applied to the polyphonic music, perform poorer ($\approx$ 20% absolute) than on the solo samples (Tian *et al.*, 2014).

In this task, a separation-driven approach for percussive onset detection is presented. A deep recurrent model (DRNN) is used to separate the percussion from the composition in the first stage. It is followed by the onset detection based on signal processing in the final stage. The proposed approach achieves significant improvement (18.4%) over the onset detection algorithm applied to the mixture and gracefully degrades (about 4.6% poorer) with respect to onset detection on solo percussion. The proposed approach has better separation and detection performance when compared to that of the baseline algorithms.

Datasets

Multi-track recordings of six live vocal concerts ($\simeq$ 14 hours) are considered for extracting the composition items. These items contain composition segments with vocal and/or violin segments in first track and percussive segments in the second track. To create the ground truth, onsets are marked (manually by the authors) in the percussive track. These onsets are verified by a professional artist[4]. Details of the datasets prepared from various concerts are given in Table 5.5. The composition items consist of recordings from both male and female artists sampled at 44.1 kHz. Some of the strokes in the mridangam are dependent on the tonic, while others are not. The concerts SS and KD also include *ghatam* and *khanjira*, which are secondary percussion instruments. Recordings are also affected by nearby sources, background applause and the perpetual *drone*.

Training examples for the percussion separation stage are obtained from the *ālāpana* (vocal solo, violin solo) and mridangam *tani āvarthanam* segments. These are mixed to create the polyphonic mixture. A total of 12 musical clips are extracted from four out of six recordings, to obtain the training set (17min and 5s), and the validation set (4min and 10s). Hence, around 43% of the data is found to be sufficient for training. 10% of the dataset is used for the validation of neural network parameters and the rest for testing the separation performance. The concert segments KK and ND are only used

[4]Thanks to musician Dr. Padmasundari for the verification

Table 5.5: Details of the dataset.

Concert	Total Length hh:mm:ss	Composition Segments mm:ss (Number)	No. of Strokes
KK	2:15:50	1:52 (3)	541
SS	2:41:14	0:38(4)	123
MH	2:31:47	1:16 (3)	329
ND	1:15:20	1:51 (3)	330
MO	2:00:15	7:14 (3)	1698
KD	2:20:23	5:32 (3)	1088
Total	13:41:59	18:23 (19)	4109

for testing the proposed approach to check the generalizability of the approach across various concerts. The composition segments shown in Table 5.5 column 3 (with ground truth) are used as the test data. Onset detection is then performed on the separated percussive track.

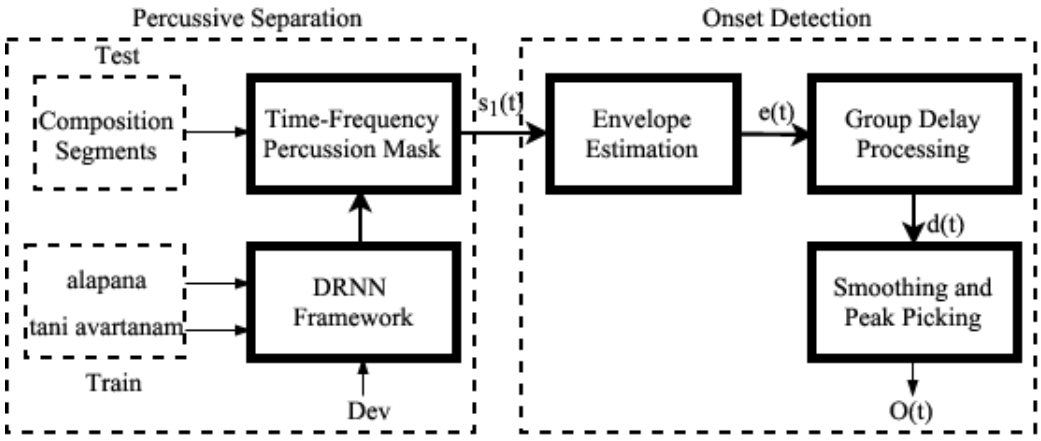


Figure 5.4: Block diagram of the proposed approach.

5.3.2 Proposed approach

The proposed method consists of two stages: percussive separation stage and solo onset detection stage. Initially, the time-frequency masks specific to percussive voices (mainly mridangam) are learned using a DRNN framework. The separated percussion source is then used as input to the onset detection algorithm. Figure 5.4 shows the block diagram of the overall process which is explained subsequently in detail.

5.3.2.1 Percussive separation stage

A deep recurrent neural network framework originally proposed for singing voice separation (Huang *et al.*, 2014*b*) is adopted for separating the percussion from the other voices. *Ālāpana* segments are mixed with *tani āvarthanam* segments for learning the

timbral patterns corresponding to each source. Figure 5.5 shows the time-frequency patterns of the composition mixture segment, melodic mixture and the percussive source in Carnatic music. The patterns associated with different voices are mixed in composition segments leading to a fairly complex magnitude spectrogram (Figure 5.5 *left*) which makes separation of percussion a nontrivial task.

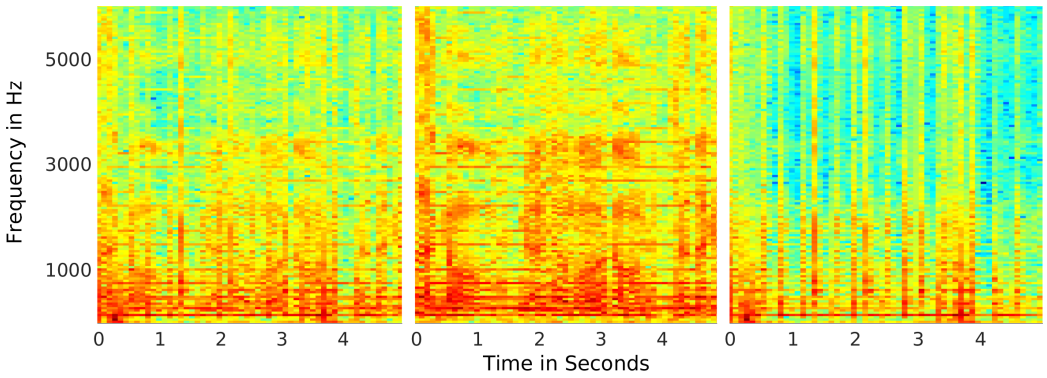


Figure 5.5: Spectrograms of a segment of composition *(left)* obtained from the mixture (KK dataset) containing melodic sources, vocal and violin *(middle)* and the percussive source *(right)*.

The DRNN architecture used for percussive separation stage is shown in Figure 5.1. The network takes the feature vector corresponding to the composition items (x_t) and estimates the mask corresponding to the percussive ($y_t^{'1}$) and non-percussive ($y_t^{'2}$) sources. The normalised mask corresponding to the percussive source ($M_1(f)$) is used to filter the mixture spectrum and then combined with the mixture phase to obtain the complex-valued percussive spectrum:

$$\widehat{S_p}(f) = M_1(f)X_t(f) \tag{5.16}$$

$$S_p(t) = ISTFT(\widehat{S_p}\angle X_t) \tag{5.17}$$

where, ISTFT refers to inverse short-time Fourier transform, $\widehat{S_p}$ is the estimated percussive spectrum, $\angle(X_t)$ is the mixture phase at time t and, $S_p(t)$ is the percussive signal estimated for t^{th} time frame.

The short-time Fourier transform (STFT) feature is used. The regression problem of finding the source specific-magnitude spectrogram is formulated as a binary mask estimation problem where each time-frequency bin is classified as either percussive or non-percussive voice. A single model is used to learn both these masks despite the fact that only percussive sound is required in the second stage. Thus, discriminative information is also used for the learning problem. The γ parameter in the objective function (Equation 5.5) is optimised such that more importance is given to minimising

the error for the percussive voices than maximising the difference with respect to the other sources. This is primarily to ensure that the characteristics of percussive voice are not affected significantly by separation, as the percussive voice will be used later for onset detection.

The recurrent connections are employed to capture the temporal dynamics of the percussive source which are not captured using the contextual windows. The network has a recurrent connection at the second hidden layer and is parametrically chosen based on the performance on development data. A recurrent network trained with *Ālāpana* and *tani āvarthanam* separates the percussion from the voice by generating a time-frequency percussive mask. This mask is used to separate the percussive voice in the composition segment of a Carnatic music item. The separated signal is used for onset detection in the next stage (Figure 5.4).

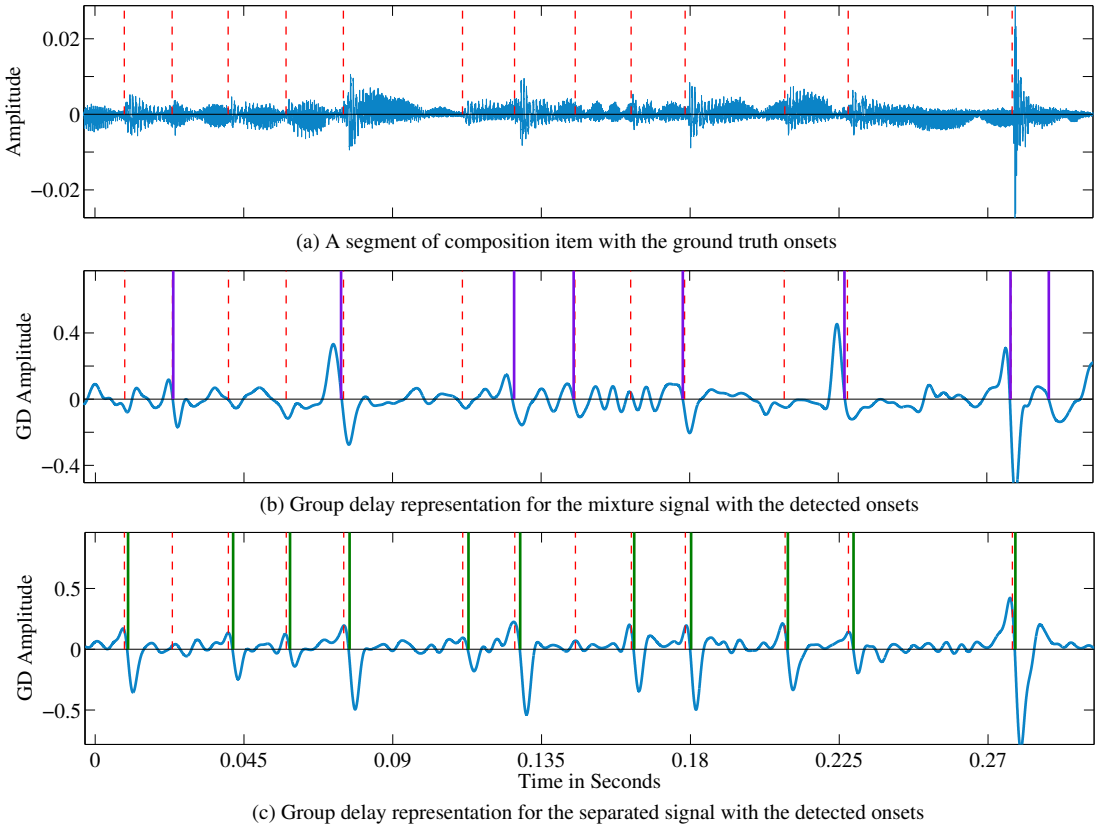
5.3.2.2 Onset detection stage

The separated percussive voice is used as the source signal for the onset detection task. Note that this signal has other source interference, artefacts and other distortions. The second block in Figure 5.4 corresponds to the onset detection stage. Onset detection consists of two steps. In the first step a detection function is derived from the percussive strokes which is then used in onset detection in the second step. For percussive onset detection, AM-FM formulation and GD based filtering technique which are discussed in Section 4.3 is used. Details of the algorithm are provided in Section 4.3.

Figure 5.6 shows an example of a composition item taken from the SS dataset. It compares the performance of the proposed approach with that of the onset detection algorithm applied directly on the mixture. Red dotted lines represent the ground truth onsets, violet (b) and green (c) lines represent the onsets detected on the mixture signal and the separated percussive signal respectively. By adjusting the threshold of onset, the number of false positives can be reduced. However, it leads to false negatives as shown in Figure 5.6 (b). The proposed approach is able to detect almost all of the actual onset locations (Figure 5.6 (c)).

5.3.3 Performance evaluation

The proposed percussive onset detection approach is developed specifically for rhythm analysis in Carnatic music composition items. However, it is instructive to compare the performance with other separation and onset detection algorithms. Also, it is important to note that the proposed approach could be applied to any music tradition with enough training musical excerpts to extract the onset locations from the polyphonic mixture. The dataset for these tasks is described in Section 5.3.1. The vocal-violin channel (*ālāpana*) and the percussion channel (*tani āvarthanam*) are mixed at 0 dB



(a) A segment of composition item with the ground truth onsets

(b) Group delay representation for the mixture signal with the detected onsets

(c) Group delay representation for the separated signal with the detected onsets

Figure 5.6: An excerpt from SS dataset illustrating the performance of the proposed approach with respect to the direct onset detection method.

SNR. The STFT with a window length of 1024 samples and hop size of 512 samples is used as the feature for training a DRNN with 3 hidden layers (1000 units/layer) and temporal connection at the 2^{nd} layer. This architecture shows a very good separation for the singing voice separation task (Huang *et al.*, 2014*b*). The dataset consists of segments with varying tempo, loudness and number of sources at a given time. The challenge lies in detecting the onsets in the presence of the interference caused by other sources and the background voices.

Evaluation metrics

Since the estimation of percussive onsets also depends on the quality of separation, it is necessary to evaluate the separated track. It is measured using three quantitative measures based on BSS-EVAL 3.0 metrics (Vincent *et al.*, 2006). The details of these measure are given in Section 5.1.

The conventional evaluation metric for the onset detection is F-measure, the details of which is explained in Section 4.3. Since it is impossible to differentiate between

simple and composite strokes for mridangam, the closely spaced onsets (within $30\ ms$) are not merged together unlike in (Böck *et al.*, 2012).

Comparison methods

The performance of the separation stage is compared with Harmonic/Percussive Separation (HPS) algorithm (Fitzgerald, 2010) for musical mixtures. It is a signal processing-based algorithm in which median filtering is employed on the spectral features for separation. Other supervised percussive separation models were specific to the musical traditions. The Non-negative Matrix Factorisation (NMF)-based approaches are not considered, since the separation performance was worse on Carnatic music. It hints the inability of a constant dictionary to capture the variability across the percussive sessions and instruments.

The onset detection performance is compared with the state-of-the-art CNN-based onset detection approach (Schlüter and Böck, 2014). In this approach, a convolutional network is trained as a binary classifier to predict whether the given set of frames has an onset or not. It is trained using percussive and non-percussive solo performances. The performance of this algorithm is evaluated on the separated percussive track and, on the mixture. The onset threshold amplitude is optimised with respect to the mixture and percussive solo channel for evaluating the performance on the separated and mixture tracks respectively for both of these algorithms.

5.3.4 Results and discussion

Percussive Separation

Table 5.6: Percussive separation performance in terms of BSS evaluation metrics for the proposed approach and HPS algorithm.

	DRNN			HPS		
Concert	GSDR	GSIR	GSAR	GSDR	GSIR	GSAR
SS	7.00	13.70	8.61	3.39	6.73	7.93
ND	7.54	17.30	8.98	0.46	3.05	7.67
KK	7.37	13.93	8.93	0.66	2.04	10.09
MH	6.40	15.64	7.63	0.82	3.31	7.79
KR	7.37	13.93	8.93	1.32	2.43	9.09
MD	6.40	15.64	7.63	2.40	8.06	4.78
Average	**7.01**	15.02	8.45	**1.50**	4.27	7.89

The results of percussive separation are compared with that of the HPS algorithm in Table 5.6. The large variability of the spectral structure with respect to the *tonic*, strokes and the percussive instruments (different types of mridangam as well) cause

the HPS model to perform poorly with respect to the proposed approach. The DRNN separation benefits from the training whereas the presence of the melodic component with rich harmonic content adds to the interference in the HPS method. This results in poor separation of the melodic mixture and percussive voice in HPS approach as indicated by an overall difference of 5.51 dB SDR with respect to DRNN approach. Although DRNN is not trained on the concerts KK and MD, separation measures are quite similar to other concerts. This is an indicator of the generalisation capability of the network since each concert is of a unique *tonic* (base) frequency, and is recorded in a different environment.

Onset detection

Table 5.7: Comparison of F-measures for the proposed approach, direct onset detection on the mixture, solo percussion channel, CNN on the mixture and on the separated percussive channel.

Concert	Proposed	Direct	Solo	CNN	CNN Sep.
SS	0.747	0.448	0.864	0.685	0.656
ND	0.791	0.650	0.924	0.711	0.740
KK	0.891	0.748	0.972	0.587	0.636
MH	0.874	0.687	0.808	0.813	0.567
KR	0.891	0.748	0.972	0.859	0.848
MD	0.874	0.687	0.808	0.930	0.919
Average	**0.845**	**0.661**	0.891	0.764	**0.727**

The accuracy of onset detection is evaluated using F-measure in Table 5.7. The performance varies with the dataset, and the results with the maximum average F-measure is reported. The degradation in performance with respect to the solo source is only about 4.6%, while the improvement in performance compared to the direct onset detection on the composite source is 18.4%. The separation step plays a crucial role in onset detection of the composition items as the performance has improved for *all* the datasets upon separation. It should be noted that the algorithm performs well for the solo percussive source. This is the reason for making comparisons with solo performances. For SS data (Table 5.5) with fast tempo (owing to multiple percussive voices) and significant loudness variation, the direct onset method causes a large number of false positives resulting in lower precision whereas the proposed approach results in a reduced number of false positives.

The proposed approach is then compared with the CNN algorithm. The optimum threshold of the solo algorithm for the Carnatic dataset (Kumar *et al.*, 2015) is used to evaluate the performance. The proposed method performs better than the CNN algorithm applied to the mixture (Table 5.7). This is because the CNN method is primarily

for solo onset detection. The performance of the baseline on the separated channel is also compared with the group delay-based method. The threshold is optimised with respect to the performance of the baseline algorithm on the mixture track. The average F-measure of the proposed approach is 11.8% better than that of the CNN-based algorithm. This is because CNN-based onset detection requires different thresholds for different concert segments. This suggests that the GD based approach generalises better in the separated voice track and is able to tolerate the inter-segment variability. A consistently better F-measure is obtained by the GD based method across all recordings. This separation-driven algorithm can be extended to any music tradition with sharp percussive onsets and having enough number of polyphonic musical ensembles for the training. These onset locations can be used to extract the strokes of percussion instruments and perform *tāla* analysis.

5.4 Hybrid Systems: 2. Gender Identification

This Section presents a raw-waveform neural network and uses it along with a denoising network for clustering in weakly-supervised learning scenarios under extreme noise conditions. Specifically, language independent Automatic Gender Recognition (AGR) is considered on a set of varied noise conditions and SNRs. The denoising problem is formulated as a source separation task and train the system using a discriminative criterion in order to enhance output SNRs. A denoising RNN is first trained on a small subset (roughly one-fifth) of the data for learning a speech-specific mask. The denoised speech signal is then directly fed as input to a raw-waveform CNN trained with denoised speech. The standalone performance of denoiser is evaluated in terms of various signal-to-noise measures and discuss its contribution towards robust AGR. An absolute improvement of 11.06% and 13.33% is achieved by the combined pipeline over the i-vector SVM baseline system for 0 dB and -5 dB SNR conditions, respectively. The information captured by the first CNN layer is further analysed in both noisy and denoised speech.

5.4.1 Introduction

Weakly-supervised learning utilises small amounts of training data, in contrast to fully supervised settings that rely on large amounts of training data (relative to test data). Such systems are particularly useful when it is possible to obtain only limited amounts of labelled data. Limited labelled data availability also challenges robust speech processing under unseen and noisy data conditions. It should be noted that most effective denoising methods in the state-of-the-art, however, are fully supervised in nature. Recent denoising algorithms use various types of neural networks for speech enhancement

as opposed to traditional signal processing-based approaches. Several variants of DNNs (Xu *et al.*, 2014; Xu, Yong *et al.*, 2015) and Denoising Auto-Encoders (DAEs) (Feng *et al.*, 2014) have been proposed for denoising the speech subject to non-stationary noise conditions. In this work, a denoising framework is presented for low-resource speech interaction applications. In particular, this work focus on the task of gender identification.

AGR from the speech signal is an essential pre-processing step for many applications and can prove to be challenging under weakly-supervised learning scenarios (Ahmad *et al.*, 2016*a*) or extreme noisy environments. Features derived from pitch and cepstral representations have been used in (Barkana and Zhou, 2015) and (Wu and Childers, 1991; Lee *et al.*, 2008; Ramdinmawii and Mittal, 2016) under clean environments. Recent DNN-based gender classification systems employ transformed MFCCs as features(Qawaqneh *et al.*, 2017). Most of the approaches are restricted to the monolingual condition. Works such as (Zeng *et al.*, 2006; Ranjan *et al.*, 2015; Harb and Chen, 2003) have however performed language-independent gender identification.

Gender identification has been performed on distorted speech in (Ranjan *et al.*, 2015) using an i-vector PLDA system, on compressed speech in (Harb and Chen, 2003) using a combination of set of experts with neural network models. Language independent AGR is performed on noisy speech in (Zeng *et al.*, 2006) with Gaussian mixture models (GMMs). This model performs well on SNRs $\geq$0 dB. However, this work does not consider challenging noisy conditions, unseen language and noise conditions during test and, the results are reported at the utterance-level by considering all the vocalised segments together using Voice Activity Detector (VAD).

Raw-waveform methods have recently been proposed for various speech processing applications such as automatic speech recognition (Palaz *et al.*, 2013; Sainath *et al.*, 2016), voice presentation attack detection (Muckenhirn *et al.*, 2017), and emotion recognition (Trigeorgis *et al.*, 2016) from speech. They are preferred due to their inherent ability to extract features specific to the application, and their superior performance. In a recent work, an end-to-end approach for gender classification, in similar lines of (Palaz *et al.*, 2013; Muckenhirn *et al.*, 2017, 2018), has been developed (Kabil *et al.*, 2018). It yielded better performance than standard acoustic features-based approach. This work builds on that work to develop a two-stage noise AGR system, where speech is denoised and then fed into the CNN for gender classification.

Language independent and weakly-supervised gender classification is performed under challenging environmental noise conditions with unseen noise and language categories in the test set. An SVM classifier is employed as the baseline system, as it provides best AGR under weakly-supervised settings (Ahmad *et al.*, 2016*a*,*b*). It uses an i-vector based feature extractor. SVM is the popular choice for classification when only a limited amount of data is available for training (Ahmad *et al.*, 2016*a*). The

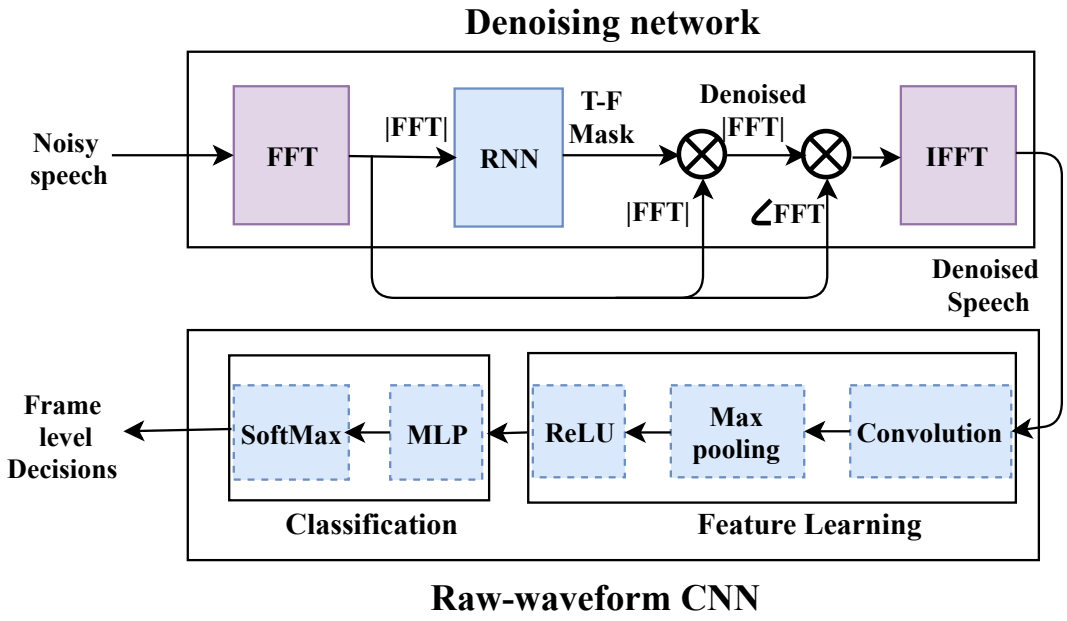


Figure 5.7: Block schematic of the proposed approach.

contributions of this work are two fold; First, it is shown that gender identification under highly-noisy conditions can be considerably improved using a denoising network. Second, the raw-waveform CNN-based approach is shown to yield significantly better results than the i-vector based approach.

5.4.2 Proposed approach

Obtaining labelled data can be time-consuming, requires skilled personnel, and is also expensive. The natural alternative is to develop unsupervised or weakly-supervised models capable of handling the variability on the test set. The latter may include differences in speaker traits such as gender and age, linguistic capabilities, and environmental factors such as noise types (e.g. stationary/non-stationary, additive/convolutive) and noise levels. Some of these issues are addressed in this method using a small fraction of the data for training and validation, and the rest for testing. The unseen noise and language conditions in our test set is simulated to investigate the robustness of the system to these conditions. Owing to these variabilities, it is vital to perform denoising as a pre-processing step. A two-stage pipeline is proposed: speech denoising stage and subsequent gender identification stage (Figure 5.7).

5.4.2.1 Denoising stage

This stage consists of three components; feature extraction, time-frequency mask estimation using denoising network and speech reconstruction. The speech denoiser is inspired by speech separation models learning both the sources simultaneously (Section 5.2). This model learns clean speech signal in its training by appropriately finding

the weights of each time-frequency bin. Noise output is obtained by considering each T-F bin by subtracting the weight-age of the clean speech from the mixture magnitude spectrogram($1 - sourceoutput$). Magnitude spectrograms of the mixture of clean speech signal ($S[n, k]$, n and k are time and frequency indices, respectively) and noise signal ($N[n, k]$) are fed to the network. The details of the network are discussed in detail in Section 5.2.

The training data is augmented by shifting either of the sources and mitigating the need for larger number of training samples. Denoised speech is obtained by multiplying the speech mask with the noisy magnitude spectrogram and using noisy phase (speech reconstruction in Figure 5.7). The patterns associated with speech are added with various background noises which lead to variabilities in the spectrogram characteristics. Figure 5.8 shows the denoising process with an example taken from the test data [5]. Weakly-supervised classification is performed in the second stage by a raw-waveform CNN on the output of the denoiser (Figure 5.7). It is argued that the classifier trained with denoised output can provide better gender identification.

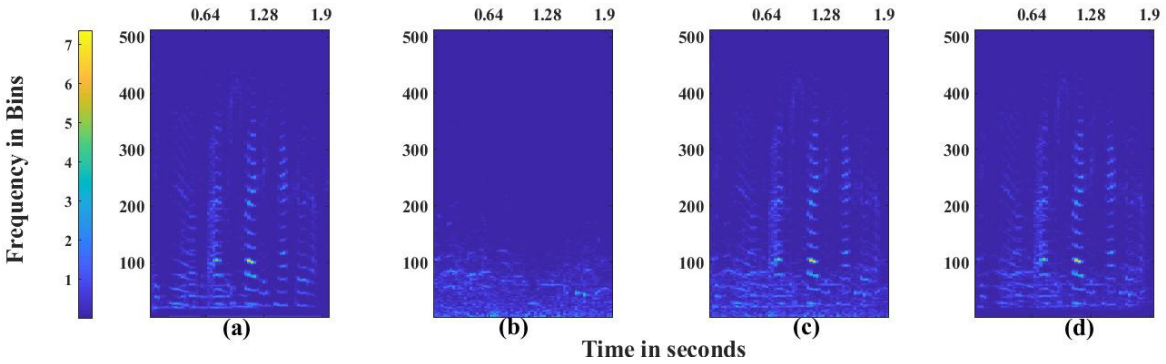


Figure 5.8: An example of denoising process: Magnitude spectrograms of *(a)* clean speech, *(b)* noise signal, *(c)* noisy speech, and *(d)* denoised speech. Observe that the denoised speech is similar to the clean one.

5.4.2.2 Classification stage: Raw waveform CNN

Similar to (Kabil *et al.*, 2018), the network consists of two sub-stages: feature learning and classification. Feature learning consists of a 1-D convolutional layer with max pooling and ReLU non-linearities, which is repeated. Classification consists of a multilayer perceptron (MLP) with ReLU activations and a softmax output. The output layer performs the softmax operation to obtain frame-level gender posteriors. The decision is made by combining the frame-level posteriors. The feature stage and classification stage are jointly trained using stochastic gradient descent algorithm with cross entropy error criterion. In (Kabil *et al.*, 2018), it was found that for effective AGR, at least two convolution layers are needed. The work considered two architectures: (a) three convo-

[5] mixture of "*scafe*" noise and female conversation in German

lution layers followed by one hidden layer, referred to as CNN1 and (b) two convolution layers followed by one hidden layer, referred to as CNN2.

Table 5.8 compares the architecture of CNN1 and CNN2. A window length of 300 ms ($\mathbf{W_{len}}$) is used with a 30 ms shift ($\mathbf{W_{shift}}$) for both architectures. In CNN1, the first convolution layer filter width is short, such that it models sub-segmental signal ($\approx$ 4 ms speech). CNN2 is used to examine the ability of raw-CNN methods with fewer parameters. This model has only $\approx$ 40% of the number of parameters compared to CNN1. CNN2 differs from CNN1 in the first convolution as it models "segmental" speech, i.e. about 20 ms speech ($\mathbf{N_{seq1}}$ = 150 samples). A max pooling size of 3 ($\mathbf{mp_i}, \mathbf{i} = \mathbf{1}..\mathbf{N} \; \forall$ convolutional layers $\mathbf{N}$) is used. The third (final) convolutional layer of CNN1 has similar dimensions as the second layer.

5.4.3 Performance evaluation

Denoising and language independent gender identification is performed on the noisy version of CALLFRIEND corpus [6]. The noise signals are selected from various categories on publicly available DEMAND (Diverse Environments Multichannel Acoustic Noise Database) corpus (Thiemann *et al.*, 2013). The following Subsections present details of the dataset, the experimental procedure, baseline system and the performance metrics.

5.4.3.1 Datasets used

The CALLFRIEND corpus consists of unscripted two channel telephonic conversation between native speakers of 13 languages. The audio is selected from the train sets of Canadian French, Farsi, Hindi, Korean and German in this work. Data are pooled such that at least two speakers from each gender are selected per language. A total of 38 speakers (19 same-gender sessions) are selected and both sides of a conversation were added together to form a two-party, one-channel recording. It ensures that long silences are not present in the recording. This corresponds to a total of 582 sessions of five minutes each. The DEMAND dataset consists of five minutes, 16 channel (microphone distance between 5 cm and 21.8 cm) environment noise recordings for 18 different noise conditions, divided into six main categories (Domestic, Nature, Office, Public, Street and, Transportation). One condition is selected from each category (*dliving, ooffice, omeeting, scafe, prestaurant, tbus*) to cover all kinds of environmental settings during the creation of noisy dataset for the experiments. One of the languages (German) and noise categories (meeting room noise) is left out for the test set during both the denoising and gender classification part, to test the robustness of the system against unseen language and environmental conditions. The noises are mixed with the conversational

[6] https://catalog.ldc.upenn.edu/

Table 5.8: Comparison of two raw-waveform architectures.

Parameters	CNN1	CNN2
number of conv. layers	3	2
L1 width/shift (in samples)/# filters	30/10/80	150/10/80
L2 width/shift (in frames)/# filters	7/1/60	7/1/60
Max pooling size/shift	3/1	3/1
number of hidden units	1024	100
Total number of parameters	433,114	184,042

speech at 0 dB and -5 dB SNRs.

5.4.3.2 Experimental procedure

Windows of 128 ms and with 64 ms shift is used to compute the short-time Fourier transform. The RNN takes and predicts a 513 point spectrum with a previous time context. It consists of a feed forward hidden layer followed by a recurrent layer, each of 500 nodes with ReLU activations. The output layer is linear. 17% of the clean and noisy data is used (93 sessions) for its training (since it is weakly-supervised), that includes 4 conversation sessions for validation. 83% of the data is used for testing. The same denoiser trained with 0 dB SNR is used for evaluating the test segments under -5 dB SNR in order to analyse its robustness. The proposed approach considers SNRs $\leq$0 dB to account for adverse environmental conditions.

The sessions are split into uniform segments of two seconds duration for classification. This is to ensure that the model is able to identify gender within a short time period. All possible combinations of noises and languages are considered with equal probability. A total of 84,240 such segments are used for the experiment. 30% of the dataset is used for training (21,494 segments) and cross-validation (3,582 segments), which includes all the training samples of the denoiser. The CNN is trained with an initial learning rate (LR) of 0.1. The LR is halved whenever the validation loss stagnates between successive epochs. Training is terminated when the LR drops below 10^{-6} and the final model is used for gender identification. The classifier is tested with 59,164 segments (70% of the dataset). Both variants of the proposed classifier (CNN1, CNN2) with a different number of hyperparameters (Table 5.8) is used.

5.4.3.3 Baseline system and performance metrics

An SVM classifier on i-vectors is used as the baseline method. SVMs are popularly chosen for learning from limited data (Ahmad *et al.*, 2016*a*). I-vectors are used as feature representation for this task. The UBM-GMM with 2048 mixtures and 400 dimensional i-vector extractor are trained using 100 sessions from the AMI meeting corpus (Carletta *et al.*, 2005) down-sampled to 8 kHz. This method provides state-of-the-art gender iden-

tification system for weakly-supervised learning (Ahmad *et al.*, 2016*a*). SVM classifier with Radial Basis Function kernel is used and the model is trained using scikit-learn python package (Pedregosa *et al.*, 2011). The effect of denoiser is analysed on the baseline as well.

Table 5.9: Denoiser performance at different noise levels.

Measure	Binary Mask		Soft Mask	
	-5 dB	0 dB	-5 dB	0 dB
GNSDR	18.09	11.12	18.24	19.50
GSIR	23.57	20.30	20.03	19.50
GSAR	14.28	13.05	14.66	14.15

Since the denoiser output is used for further processing, its metrics should be able to accommodate both the amount of noise and artefacts introduced in the denoising process as opposed to the traditional evaluation metrics (SNR and PESQ). signal to interference ratio (SIR), signal to artefacts ratio (SAR) and signal to distortion ratio (SDR) from BSS Evaluation Metrics (Vincent *et al.*, 2006) are chosen for evaluation. Unweighted average recall (UAR) is reported for gender classification since it is robust to class imbalance.

5.4.4 Results and discussions

The performance of denoiser is reported in Table 5.9. Both binary and soft masks are used in the experiments and could observe that binary mask performs better compared to soft mask in general. The denoiser is trained at 0 dB mixing condition and tested on both 0 dB and -5 dB conditions. Discriminative training causes larger SIR values (Huang *et al.*, 2015). Denoiser performs equally well for unseen noise category (*omeeting*), language (*German*) and their combinations. The denoiser has all evaluated metrics above 10 dB. Its role in gender classification is further analysed.

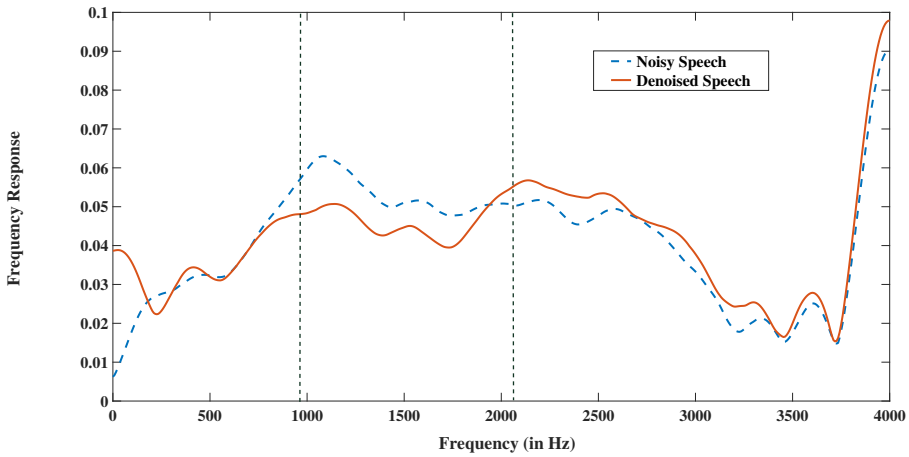


Figure 5.9: Cumulative Frequency Response of Layer1 filters in CNN1.

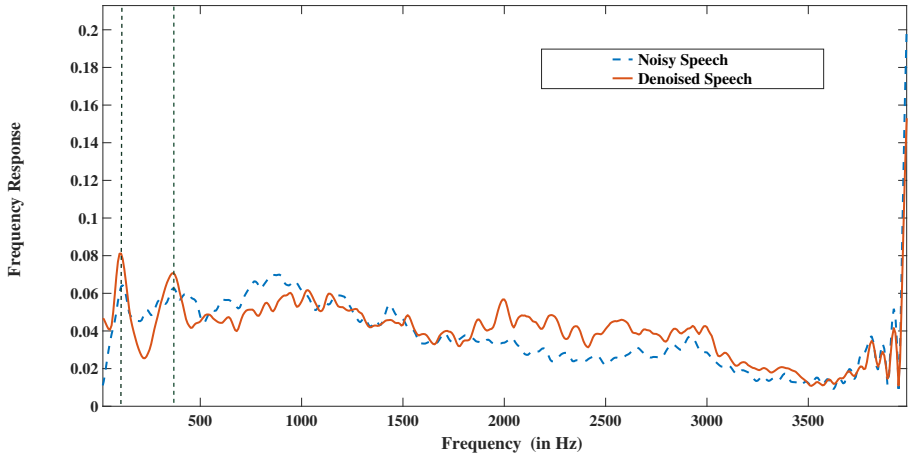


Figure 5.10: Cumulative Frequency Response of Layer1 filters in CNN2.

The results of language independent AGR are shown in Table 5.10. Systems trained with noisy speech at 0 dB SNR and its denoised version are used directly for testing the noisy speech at -5 dB SNR and its denoised version respectively. All systems show a consistent improvement over the baseline under both noise levels. Raw-waveform CNN architectures perform significantly better than the baseline. As expected, the UAR is higher for 0 dB as compared to -5 dB mixing condition across the architectures. Denoiser improves the performance of *all* of them. An absolute improvement of 11.06% and 13.33% is achieved by a combination of denoiser and raw CNN method over the baseline for 0 and -5 dB SNRs respectively.

Table 5.10: Gender identification performance in terms of UAR (%) at different noise levels.

System	Noisy		Denoised	
	-5 dB	0 dB	-5 dB	0 dB
Baseline	76.84	81.95	79.83	83.34
CNN2	83.86	89.13	88.00	91.53
CNN1	87.47	91.31	**90.17**	**93.01**

Cumulative Frequency Response (CFR) of the learned filters is shown in Figure 5.10. It is obtained by normalising the sum of all the filter responses (Muckenhirn *et al.*, 2018) and shows the frequency regions the filters emphasise collectively. The filters learned from the noisy speech and denoised speech are *similar*, except that the denoised versions provide room for a clearer analysis. CNN1 seems to give emphasis to formant regions, around 1000 and 2000Hz whereas, CNN2 captures gender discriminative information in low frequency regions as well as high frequency regions. Specifically, CNN2 CFR has two peaks at 101 Hz and 351 Hz, potentially modelling male and female average fundamental frequency respectively. These observations indicate that CNN with different architectures learns to weigh the frequency spectrum at different resolutions - capturing vocal tract information in one (CNN1) and fundamental

frequency in another (CNN2).

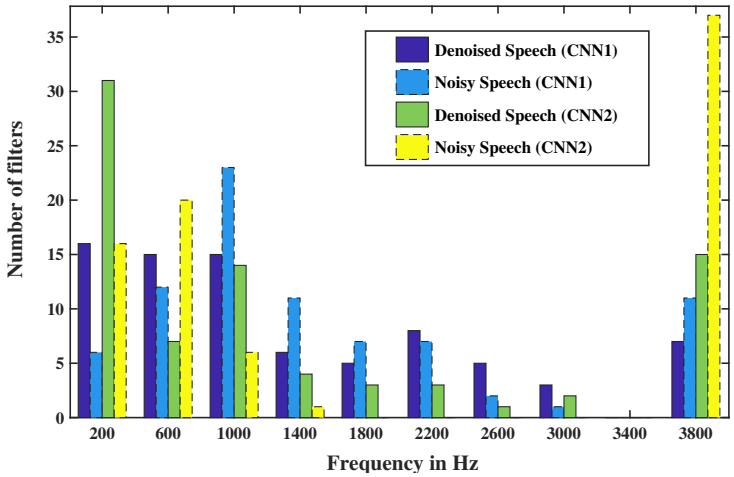


Figure 5.11: Histograms of peak frequency responses of filters in sorted order.

The histograms of peak frequency responses of filters in sorted order are shown in Figure 5.11. There are larger number of peak frequency filters in low-frequency region ($\leq$ 400 Hz) for the denoised speech as compared to noisy speech, possibly due to pitch and low-frequency formants easier to learn in denoised conditions. The plot also reveals the frequency selective nature of individual filters. Observe that, there are more number of filters with peak frequency in the high-frequency region in noisy speech than that in denoised speech. This could be due to the artefacts and presence of high-frequency noise. Further analyses inclusive of original clean speech models are required to understand them.

5.5 Summary

This chapter discussed various systems used for source separation. An overview to source separation is followed by the proposed approaches for musical source separation. Discriminative ability of group delay is exploited for singing voice separation and vocal-violin separation tasks. The effectiveness of source separation framework exploited in this task is further used for percussive separation and speech denoising tasks. These pre-processing stages are found to be effective in onset detection from composition segments of Carnatic music and also in automatic gender recognition system under noisy and limited data conditions. Singing voice separation using adapted S2S neural network is proposed which learns task-specific features. This chapter thus illustrate the effectiveness of data-driven methods for source separation applications and shows that signal processing can still contribute to the performance of signal extraction and detection systems.

CHAPTER 6

Conclusions

The tasks of estimating the instances of temporal significance in a signal and, extracting the signal of relevance from a mixture signal are considered in this thesis. These tasks are traditionally solved using signal processing or machine learning or a combination of both. For this thesis, these tasks are motivated by the importance of phase-based processing and further its use as a feature in supervised methods. End-to-end systems presented in this thesis are inspired owing to their success over the feature-based approaches for other speech processing tasks.

A generalised mathematical proof governing the high resolution property of group delay functions is provided first. Particularly, this thesis extends the HR property for multi pole minimum phase systems. Inspired by this property, algorithms for various time-event detection (TED) tasks are performed, namely, pitch estimation from speech using grating compression transform on modified-group delay-gram, percussive onset detection from Carnatic percussion instruments and spike estimation from Calcium fluorescence signals. Ability of GD as a feature to distinguish timbre characteristics are investigated for source separation tasks. Musical source separation is performed using modified group delay as the feature representation. This separation network is then used as a pre-processing stage for percussive onset identification from musical mixture and for automatic gender identification from noisy speech signals. Finally, a signal-to-signal (s2s) conversion neural network architecture is proposed for spike estimation and singing voice separation tasks.

The GD is agnostic to the signal at hand. This makes it possible to apply GD processing for any type of signals. Pitch estimation using group delay is not novel as such. However, using it in conjunction with an established pitch estimation such as GCT made the algorithm superior to the basic pitch estimation algorithms and comparable to popular approaches in the field. Solo percussive onset detection algorithm uses an AM-FM formulation suitable for percussive signals. The pre-processed signal is directly used for estimating the onset locations using GD processing.

Applicability of GD beyond speech and music signals is exploited for spike estimation task in which the GD acts as a local peak enhancer. The non model based algorithm performs in comparison with the established approaches for spike estimation. Proposed end-to-end neural net for spike estimation is inspired by its success in other speech tasks. Each layer of S2S contributes to the global aim of spike information extraction. It provides state of the art results with an additional scope to analyse the layer-wise output and the responses of the learned filters.

Source separation using MODGDgram feature instead of spectrogram provides improvements in SIR or a reduction in complexity of the network for a similar performance. The basis functions could be learned directly from the data using an S2S adapted for the singing voice separation. Separation system can also be used as front-end for detection tasks in mixed conditions. The performance of the system is proportional to the performance of the separation stage.

The major contributions of the thesis are as follows:

- Mathematical proofs governing the high-resolution property of group delay functions are extended for multi pole systems.
- Relevance of group delay is studied for applications related to time-events. Proposed algorithms for pitch estimation from speech, percussive onset detection from music signals and spike estimation from neuronal signals.
- An end-to-end neural network is proposed for spike estimation task.
- A new feature is proposed for musical source separation based on modified group delay processing, exploiting the discriminative ability of group delay.
- Inspired by the success of TED and source separation tasks, hybrid systems are proposed for percussive onset detection from composition items of Carnatic music and language independent automatic gender recognition from noisy speech.
- Extended the ability of S2S to singing voice separation to learn source-specific masks in an end-to-end manner.

6.1 Criticisms

- An analysis of HR property for minimum phase systems is provided. However, an explicit expression for the n dB bandwidth is not provided for the multi pole systems.
- The performance of proposed pitch estimation algorithm is not as robust as the advanced algorithms since the erroneous continuous contours on the MODGD are likely to be propagated to the second stage of the algorithm.
- Percussive onset detection fails to accurately pick the soft onsets since the scope of AM-FM modelling is limited to percussive instruments.
- GDspike algorithm does not take into account the delay between the spike occurrence and the visible fluorescence change. This delay could be significant as large as 1 sec for some indicators.

- Simple triangulation step of the GDspike algorithm causes a spike information signal with several low-valued peaks, which degrades the performance measures computed directly on the analogue estimates of the spikes.
- The performance of S2S neural network proposed for spike estimation algorithm degrades when a different dataset without pre-processing (to remove the linear trends) is used only for testing. S2S need to be trained at least with a subset of the new dataset to obtain the state-of-the-art performance.
- MODGDgram feature does not improve the performance over magnitude spectrum based features for speech source separation. This is because the harmonic and rhythmic structure (clear peaks and valleys in the spectrum/GD function) is not very prominent in speech. Multi-speaker speech separation requires the model to be independent of the speaker. Magnitude spectrum has a better smoothing capability across different speakers than the MODGDF.
- MODGD based feature often causes more artefacts (though of negligible difference) than magnitude spectrum. MODGD has sharper time-frequency values and when they become erroneous, introduce artefacts at the signal output.
- S2S based singing voice separation requires more running time than feature-based counterparts, owing to the filters which needs to be learned. The filter initialisation is crucial for convergence, especially when the loss function does not directly reflect the separation quality.
- The errors in the pre-processing stage is propagated to the detection stages for both of the hybrid systems presented. A separated signal with a large SIR does not guarantee better detection/classification as it suffers from artefacts which could adversely affect the feature selection/learning of the second stage.

6.2 Future Directions

The following are the possible future directions:

- GD based processing could be adopted for other time-event detection tasks such as glottal closure instants (GCI) from speech or sharpening the high-energy locations of a waveform for improved estimation of these positions.
- Multi-instrument separation from musical mixture is a source separation task which can be performed using MODGDgram as the mixture shows patterns of individual instruments better than that of magnitude spectrum. This improves the visibility of GD-based musical separation.

- The effectiveness of S2S can be exploited in other time-event detection tasks such as percussive and general onset detection, single and multi-pitch estimation, epoch extraction from speech signals etc. This can also be used in signal extraction tasks such as speech denoising and for hybrid systems such as onset detection from composition mixtures.

- Another potential application of S2S is that it can be used for signal-to-signal transformation applications such as voice conversion. The proposed architecture could also be adopted for computational brain research applications such as prediction of electroencephalograph (EEG) from speech and other input stimuli.

- Percussive onset detection from a mixed musical segment can be performed in an end-to-end fashion which alleviate the need of the separate separation framework. A joint training could be implemented for both of the signal extraction based hybrid systems proposed in this thesis.

APPENDIX A

Composition Items in Carnatic Music

Carnatic music concerts consists of segments of similar structure known as item. It has an optional pure melodic section (*ālāpana*), followed by a mandatory lyrical composition section and an optional percussion section in which only the percussive instruments are played (*tani āvarthanam*). An item in a vocal Carnatic music concert can have different structures and it usually ends with a composition segment. A concert lasts for about 23 hours long and generally contains 8 to 10 items.

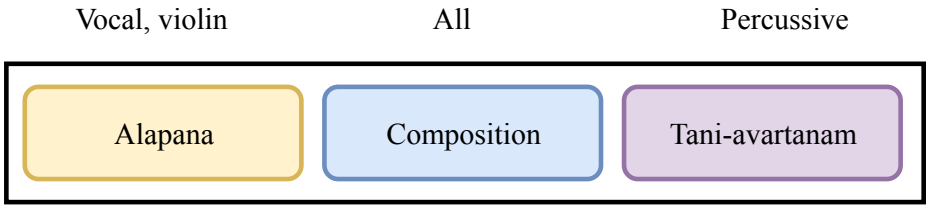


Figure A.1: Schematic of a Carnatic music item.

In an *ālāpana*, the vocalist/violinist elaborates the *rāgā* based on her/his *manodharma* or imagination. The composition is rendered by the vocal artist with accompaniments including harmonic and percussion instruments. The composition is set to the *tālā* and *rāgā* which are defined by the composer, but based on the tonic of the artist. Some sections in the compositions can be improvised as well. Composition is an important segment in a concert and it occurs in every item. In a composition segment, all the instruments are played together (Figure A.1). A *tani āvarthanam* segment is rendered by one or more percussive artists with percussive instruments such as the mridangam, the ghatam, the thavil, the morsing and/or the kanjira.

APPENDIX B

Review of percussion instruments in Carnatic music

Mridangam

Mridangam is the primary percussion instrument in Carnatic music. It resembles a two-sided drum with tightly stretched membranes on either side, with two unequal sized sides.

The instrument is tuned by loading the smaller side with a black paste of adhesive mixed with finely ground particular stone. In this context, the tonic is defined as the base pitch, which is used as a reference for all other higher harmonics. The strokes can be categorised based on the side of the mridangam being played and the position, manner and force with which the membranes are struck. However, the exact number of unique strokes varies across different schools of mridangam. The two sides allow composite strokes (individual strokes from the left and right side at the same instant) to be created which from an MIR perspective ought to be treated as one, although they sometimes appear as separate strokes while performing the onset detection analysis. The first study on mridangam carried out by Nobel prize-winning scientist Raman (Raman, 1934) and later by Siddharthan (Siddharthan *et al.*, 1994) analysed the harmonics of the strokes. More recently, Akshay (Anantapadmanabhan *et al.*, 2013) employed non-negative matrix factorisation to classify the strokes.

Ghatam

The ghatam is a hollow pot that is placed on the lap of the artist and struck with the palm and fingers. The instrument is made of specifically burnt clay with metallic powder for strength and care is taken that the walls of ghatam are of equal thickness. Distinct ghatam strokes count lesser in number than mridangam. Tuning of the pitch is possible to a limited extent by application of *play-doh*, but mostly another ghatam is chosen to achieve significant variations. Ghatam strokes also produce a characteristic sound when struck on the neck of the pot. Finally, the artist modulates the sound by modifying the size of the mouth during the performance, by partly or wholly closing the area of the mouth with palms.

Morsing

Known as *Jew's Harp*, the morsing is a wind percussion instrument. It resembles a metallic clamp with a thin film (the *tongue*) in between them. The instrument is caught by hand and placed in the mouth of the artist, the teeth firmly holding it in place. The sound is produced inside the mouth of the artist by triggering the tongue of the instrument with the index finger. The artist's tongue is also used to produce morsing notes. Pitch of the instrument cannot be varied significantly, and the artists prefer to carry morsings of different dimensions for fine-tuning.

Thavil

The thavil is similar to the mridangam in the sense that it is a two-sided barrel, with both sides participating in the sound production. The left side is struck with a stick while the artist plays the right side with fingertips covered with *thumb caps*. The *thumb caps* are mostly made of hardened rice flour and give rise to sharp, cracking sounds. Variations in pitch are attained by tightening the left side of the instrument. Distinct strokes exist, based on the side of the instrument struck and the number of fingers involved in the production (for the right side). For instance, *Ta* and *Di* involve four fingers but are still treated as a single stroke by musicians.

Kanjira

The kanjira is a one-sided percussion instrument and is small enough to be held with one hand. The instrument is made of monitor lizard belly skin stretched across a circular wooden frame made from the jackfruit tree. High pitched sound is produced by striking the circular face with the palm and fingers of the free hand. Unlike the mridangam, the face of the kanjira is not loaded with any paste. The pitch can be varied to an extent by applying pressure on the face using the hand holding the kanjira or by spraying water on the kanjira skin from behind.

References

1. **Abadi, M.**, **A. Agarwal**, *et al.* (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

2. **Ahmad, J.**, **M. Fiaz**, **S.-i. Kwon**, **M. Sodanil**, **B. Vo**, and **S. W. Baik** (2016*a*). Gender identification using mfcc for telephone applications-a comparative study. *arXiv preprint arXiv:1601.01577*.

3. **Ahmad, J.**, **K. Muhammad**, **S.-i. Kwon**, **S. W. Baik**, and **S. Rho**, Dempster-shafer fusion based gender recognition for speech analysis applications. *In Platform Technology and Service (PlatCon), 2016 International Conference on*. IEEE, 2016*b*.

4. **Akerboom, J.** *et al.* (2012). Optimization of a GCaMP calcium indicator for neural activity imaging. *The Journal of neuroscience*, **32**(40), 13819–13840.

5. **Anantapadmanabhan, A.**, **A. Bellur**, and **H. A. Murthy**, Modal analysis and transcription of strokes of the mridangam using non-negative matrix factorization. *In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. 2013.

6. **Asha, T.**, **M. Saranya**, **D. K. Pandia**, **S. Madikeri**, and **H. A. Murthy**, Feature switching in the i-vector framework for speaker verification. *In Fifteenth Annual Conference of the International Speech Communication Association*. 2014.

7. **Barkana, B. D.** and **J. Zhou** (2015). A new pitch-range based feature set for a speakers age and gender classification. *Applied Acoustics*, **98**, 52–61.

8. **Bastys, A.**, **A. Kisel**, and **B. Salna** (2010). The use of group delay features of linear prediction model for speaker recognition. *INFORMATICA*, **21**, 1–12.

9. **Bay, M.** and **J. W. Beauchamp** (2007). Multi-f0 tracking & harmonic source separation. *Music Information Retrieval Evaluation eXchange (MIREX)*.

10. **Bello, J. P.**, **L. Daudet**, **S. Abdallah**, **C. Duxbury**, **M. Davies**, and **M. B. Sandler** (2005). A tutorial on onset detection in music signals. *Speech and Audio Processing, IEEE Transactions on*, **13**(5), 1035–1047.

11. **Bello, J. P.**, **C. Duxbury**, **M. Davies**, and **M. Sandler** (2004). On the use of phase and energy for musical onset detection in the complex domain. *IEEE Signal Processing Letters*, **11**(6), 553–556.

12. **Bello, J. P.** and **M. Sandler**, Phase-based note onset detection for music signals. *In Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 5. IEEE, 2003.

13. **Bellur, A.** (2013). *Automatic Tonic Identification in Indian Classical Music*. M.S.thesis, Indian Institute of Technology, Department of Electrical Engg., Madras, India.

14. **Bellur, A.**, **V. Ishwar**, **X. Serra**, and **H. A. Murthy**, A knowledge based signal processing approach to tonic identification in indian classical music. *In International CompMusic Wokshop*. 2012.

15. **Bellur, A.** and **H. A. Murthy** (2013*a*). A novel application of group delay function for identifying tonic in carnatic music. *Proceedings of 21st Europian Signal Processing Conference, Marrakech, Morocco*, **ISBN -978-1-4799-3687-8**.

16. **Bellur, A.** and **H. A. Murthy**, A novel application of group delay functions for tonic estimation in carnatic music. *In eusipco*. 2013*b*. ISSN 2219-5491.

17. **Belouchrani, A.** and **M. G. Amin**, New approach for blind source separation using time-frequency distributions. *In SPIE's 1996 International Symposium on Optical Science, Engineering, and Instrumentation*. International Society for Optics and Photonics, 1996.

18. **Benetos, E.** and **S. Dixon**, Polyphonic music transcription using note onset and offset detection. *In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011.

19. **Berens, P.**, **J. Freeman**, **T. Deneux**, **N. Chenkov**, **T. McColgan**, **A. Speiser**, **J. H. Macke**, **S. C. Turaga**, **P. Mineault**, **P. Rupprecht**, *et al.* (2018). Community-based benchmarking improves spike rate inference from two-photon calcium imaging data. *PLoS computational biology*, **14**(5), e1006157.

20. **Berkhout, A. J.** (1974). Related propeties of minimum phase and zero phase time functions. *Geophysical Prospecting*, 683–709.

21. **Böck, S.**, **F. Krebs**, and **M. Schedl**, Evaluating the online capabilities of onset detection methods. *In Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2012)*. 2012.

22. **Böck, S.** and **G. Widmer**, Maximum filter vibrato suppression for onset detection. *In In Proc. Digital Audio Effects Workshop (DAFx*. 2004.

23. **Böck, S.** and **G. Widmer**, Local group delay based vibrato and tremolo suppression for onset detection. *In Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013)*. Curitiba, Brazil, 2013.

24. **Boersma, P.** (2001). Praat, a system for doing phonetics by computer. *Glot International*, **5**(9/10), 341–345.

25. **Boulanger-Lewandowski, N.**, **G. J. Mysore**, and **M. Hoffman**, Exploiting long-term temporal dependencies in nmf using recurrent neural networks with application to source separation. *In Acoustics, Speech and Signal Processing (ICASSP), 2014*. IEEE, 2014.

26. **Bozkurt, B.**, **L. Couvreur**, and **T. Dutoit** (2007). Chirp group delay analysis of speech signals. *Speech Communication*, **49**(3), 159–176.

27. **Cardoso, J.-F.** (1998). Blind signal separation: statistical principles. *Proceedings of the IEEE*, **86**(10), 2009–2025.

28. **Carletta, J.**, **S. Ashby**, **S. Bourban**, **M. Flynn**, **M. Guillemot**, **T. Hain**, **J. Kadlec**, **V. Karaiskos**, **W. Kraaij**, **M. Kronenthal**, *et al.*, The ami meeting corpus: A pre-announcement. *In International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005.

29. **Chan, W.**, **N. Jaitly**, **Q. Le**, and **O. Vinyals**, Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. *In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016.

30. **Chandna, P.**, **M. Miron**, **J. Janer**, and **E. Gómez**, Monoaural audio source separation using deep convolutional neural networks. *In International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2017.

31. **Chen, T.-W.** *et al.* (2013*a*). Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, **499**(7458), 295–300.

32. **Chen, T.-W.** *et al.* (2013*b*). Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, **499**(7458), 295–300. ISSN 1476-4687. URL `http://dx.doi.org/10.1038/nature12354`.

33. **Cheveigne, A. D.** and **H. Kawahara** (2002.). Yin, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Amer.*, 111(4):1917–1930.

34. **Chiu, C.-C.**, **T. N. Sainath**, **Y. Wu**, **R. Prabhavalkar**, **P. Nguyen**, **Z. Chen**, **A. Kannan**, **R. J. Weiss**, **K. Rao**, **E. Gonina**, *et al.*, State-of-the-art speech recognition with sequence-to-sequence models. *In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

35. **Chollet, F.** *et al.* (2015). Keras. `https://github.com/fchollet/keras`.

36. **Cotton, R. J.**, **E. Froudarakis**, **P. Storer**, **P. Saggau**, and **A. S. Tolias** (2013). Three-dimensional mapping of microcircuit correlation structure. *Frontiers in neural circuits*, **7**, 151.

37. **Dana, H.** *et al.* (2016). Sensitive red protein calcium indicators for imaging neural activity. *eLife*, **5**, e12727. ISSN 2050-084X. URL `https://elifesciences.org/content/5/e12727`.

38. **DeCarlo, L. T.** (1997). On the meaning and use of kurtosis. *Psychological methods*, **2**(3), 292.

39. **Deivapalan, P.**, **M. Jha**, **R. Guttikonda**, and **H. A. Murthy** (2008). DONLabel: An automatic labeling tool for indian languages,. *Proceedings of National Conference on Communication*, 263–266.

40. **Deneux, T.** *et al.* (2016). Accurate spike estimation from noisy calcium signals for ultrafast three-dimensional imaging of large neuronal populations in vivo. *Nature Communications*, **7**(July), 12190. ISSN 2041-1723. URL `http://www.nature.com/doifinder/10.1038/ncomms12190`.

41. **Dey, S.**, **R. Rajan**, **R. Padmanabhan**, and **H. Murthy** (2011). Feature diversity for emotion, language and speaker verification. *Proceedings of National Conference on Communication(NCC),Indian Institute of Sciences, Banglore*, 1 – 5.

42. **Diment, A.**, **E. Cakir**, **T. Heittola**, and **T. Virtanen** (2016). Automatic recognition of environmental sound evenst using all-pole group delay features. *Proceedings of 23rd EUSIPCO*, 734–738.

43. **Dixon, S.**, Onset detection revisited. *In Proc. of the 9th Int. Conference on Digital Audio Effects (DAFx06)*. 2006.

44. **Duan, Z.**, **Y. Zhang**, **C. Zhang**, and **Z. Shi** (2008). Unsupervised single-channel music source separation by average harmonic structure modeling. *Audio, Speech, and Language Processing, IEEE Transactions on*, **16**(4), 766–778.

45. **Dubnov, S.** (2004). Generalization of spectral flatness measure for non-gaussian linear processes. *Signal Processing Letters, IEEE*, **11**(8), 698–701.

46. **Duncan, G.**, **H. A. Murthy**, and **B. Yegnanarayana** (1989). A nonparametric method of formant estimation using group delay spectra. *ICASSP*, **1**, 572–575.

47. **Duxbury, C.**, **J. P. Bello**, **M. Sandler**, **M. S**, and **M. Davies**, A comparison between fixed and multiresolution analysis for onset detection in musical signals. *In In Proc. Digital Audio Effects Workshop (DAFx*. 2004.

48. **Every, M. R.** and **J. E. Szymanski** (2006). Separation of synchronous pitched notes by spectral filtering of harmonics. *Audio, Speech, and Language Processing, IEEE Transactions on*, **14**(5), 1845–1856.

49. **Eyben, F.**, **S. Böck**, **B. Schuller**, and **A. Graves** (2010). Universal onset detection with bidirectional long short-term memory neural networks. *In Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2010)*, 589–594.

50. **Feng, X.**, **Y. Zhang**, and **J. Glass** (2014). Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014*, 1759–1763.

51. **Fitzgerald, D.** (2010). Harmonic/percussive separation using median filtering. *In Proceedings of the 13th International Conference on Digital Audio Effects (DAFx)*, 15–19.

52. **Fitzgerald, D.**, **A. Liukus**, **Z. Rafii**, **B. Pardo**, and **L. Daudet** (2014). Harmonic/percussive separation using kernel additive modelling. *In Proceedings of the 25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014)*, 35–40.

53. **Friedrich, J.** and **L. Paninski** (2016). Fast active set methods for online spike inference from calcium imaging. *Advances In Neural Information Processing Systems*, 1984–1992.

54. **Fu, S.-W.**, **Y. Tsao**, **X. Lu**, and **H. Kawai** (2017). Raw waveform-based speech enhancement by fully convolutional networks. *arXiv preprint arXiv:1703.02205*.

55. **Gabor, D.** (1946). Theory of communication. *The Journal of the Institution of Electrical Engineers*, **93**(26), 429–457.

56. **Gabrielli, L.**, **F. Piazza**, and **S. Squartini** (2011). Adaptive linear prediction filtering in dwt domain for real-time musical onset detection. *EURASIP Journal on Advances in Signal Processing*.

57. **Glorot, X.**, **A. Bordes**, and **Y. Bengio** (2011). Deep sparse rectifier neural networks. *In International Conference on Artificial Intelligence and Statistics*, 315–323.

58. **Golda, B. R.** and **H. A. Murthy** (2013). Analysis of vowel deletion in continuous speech. *In Proceedings of EUSIPCO-2013*, Th–L2.6. ISSN 2219-5491.

59. **Gonzalez, S.** and **M. Brookes** (2014). Pefac-a pitch estimation algorithm robust to high levels of noise. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, **22**(2), 518–530.

60. **Goto, M.** and **Y. Muraoka** (1994). A sound source separation system for percussion instruments. *Transactions of the Institute of Electronics, Information and Communication Engineers*, **77**, 901–911.

61. **Goto, M.** and **Y. Muraoka** (1996). Beat tracking based on multiple-agent architecture a real-time beat tracking system for audio signals. *In Proceedings of Second International Conference on Multiagent Systems*, 103–110.

62. **Greenberg, D. S.**, **A. R. Houweling**, and **J. N. Kerr** (2008). Population imaging of ongoing neuronal activity in the visual cortex of awake rats. *Nature neuroscience*, **11**(7), 749–751.

63. **Greenberg, S.**, **S. Chang**, and **J. Hollenback** (2000). An introduction to the diagnostic evaluation of switchboard corpus automatic speech recognition systems. *In Proceedings of NIST Speech Transcription Workshop*.

64. **Grewe, B. F.**, **D. Langer**, **H. Kasper**, **B. M. Kampa**, and **F. Helmchen** (2010). High-speed in vivo calcium imaging reveals neuronal network activity with near-millisecond precision. *Nature methods*, **7**(5), 399–405.

65. **Handel, S.** (1989). Listening: An introduction to the perception of auditory events. *MIT Press*.

66. **Harb, H.** and **L. Chen** (2003). Gender identification using a general audio classifier. *In Proceedings of International Conference on Multimedia and Expo, ICME 2003*, **2**, II–733.

67. **Hari Krishnan P, S.**, **R. Padmanabhan**, and **H. A. Murthy** (2006). Robust voice activity detection using group delay functions. *In Proceedings of IEEE International Conference on Industrial Technology, Mumbai, India*, 2603–2607.

68. **Hinton, G.**, **L. Deng**, **D. Yu**, **G. E. Dahl**, **A.-r. Mohamed**, **N. Jaitly**, **A. Senior**, **V. Vanhoucke**, **P. Nguyen**, **T. N. Sainath**, *et al.* (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, **29**(6), 82–97.

69. **Holzapfel, A.**, **Y. Stylianou**, **A. Gedik**, and **B. Bozkurt** (2010). Three dimensions of pitched instrument onset detection. *IEEE Transactions on Audio, Speech, and Language Processing*, **18**(6), 1517–1527. ISSN 1558-7916.

70. **Hsu, C.-L.** and **J.-S. R. Jang** (2010). On the improvement of singing voice separation for monaural recordings using the mir-1k dataset. *IEEE Transactions on Audio, Speech, and Language Processing*, **18**(2), 310–319.

71. **Huang, P.-S.**, **S. D. Chen**, **P. Smaragdis**, and **M. Hasegawa-Johnson** (2012). Singing-voice separation from monaural recordings using robust principal component analysis. *Acoustics, Speech and Signal Processing (ICASSP), 2012*, 57–60.

72. **Huang, P.-S.**, **M. Kim**, **M. Hasegawa-Johnson**, and **P. Smaragdis** (2014*a*). Deep learning for monaural speech separation. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014*, 1562–1566.

73. **Huang, P.-S.**, **M. Kim**, **M. Hasegawa-Johnson**, and **P. Smaragdis** (2014*b*). Singing-voice separation from monaural recordings using deep recurrent neural networks. *IN Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*.

74. **Huang, P.-S.**, **M. Kim**, **M. Hasegawa-Johnson**, and **P. Smaragdis** (2015). Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **23**(12), 2136–2147.

75. **Humble, M.** (2002). The development of rhythmic organization in indian classical music. *MA dissertation, School of Oriental and African Studies, University of London.*, 27–35.

76. **Janakiraman, R.**, **C. K. J.**, and **M. H. A.**, Robust syllable segmentation its application to syllable-centric continuous speech recognition. *In Proceedings of National Conference on Communications*. 2010.

77. **Jayant, N. S.** and **P. Noll** (1984). Digital coding of waveforms, principles and applications to speech and video. *Prentice-Hall, Englewood Cliffs NJ, USA*, 688. N. S. Jayant: Bell Laboratories; ISBN 0-13-211913-7.

78. **Jayesh, M.** and **C. Ramalingam** (2014). An improved chirp group delay based algorithm for estimating the vocal tract response. *In Proceedings of European Signal Processing Conference (EUSIPCO)*, 2295–2299.

79. **Jayesh, M.** and **C. Ramalingam** (2016). Improved chirp group delay based algorithms with applications to vocal tract estimation and speech recognition. *Speech Communication*, **81**, 72–89.

80. **Johnston, J. D.** (1988). Transform coding of audio signals using perceptual noise criteria. *Selected Areas in Communications, IEEE Journal on*, **6**(2), 314–323.

81. **Jung, T.-P.**, **S. Makeig**, **C. Humphries**, **T.-W. Lee**, **M. J. Mckeown**, **V. Iragui**, and **T. J. Sejnowski** (2000). Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, **37**(2), 163–178.

82. **K., P. V.**, **N. T.**, and **H. A. Murthy** (2004). Automatic segmentation of continuous speech using minimum phase group delay functions. *Speech Communications*, **42**, pp.429–446.

83. **Kabil, S. H.**, **H. Muckenhirn**, and **M. Magimai.-Doss** (2018). On learning to identify genders from raw speech signal using CNNs. *Proceedings of INTERSPEECH 2018 – 19th Annual Conference of the International Speech Communication Association, September 2-6, Hyderabad, India*.

84. **Kasi, K.** and **S. A. Zahorian** (2002). Yet another algorithm for pitch tracking. *IN Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2002*, **1**, I–361–I–364. ISSN 1520-6149.

85. **Kass, R. E.** and **V. Ventura** (2001). A spike-train probability model. *Neural computation*, **13**(8), 1713–1720.

86. **Kumar, J. C.**, **R. Janakiraman**, and **H. A. Murthy** (2010). KL divergence based feature switching in the linguistic search space for automatic speech recognition. *In Proceedings of National Conference on Communication, Indian Institute of Technology Madras*, 1–5.

87. **Kumar, J. C.** and **H. A. Murthy** (2009). Entropy based measures for incorporating feature stream diversity in the linguistic search space for syllable based automatic annotated recognizer. *In Proceedings of National Conference on Communication, Indian Institute of Technology, Guwahati*, 286–289.

88. **Kumar, M.**, **J. Sebastian**, and **H. A. Murthy** (2015). Musical onset detection on carnatic percussion instruments. *In Proceedings of Twenty First National Conference on Communications (NCC), 2015*, 1–6.

89. **Kumar, P.** (2015). *High Resolution Property of Group Delay and its Application to Musical Onset Detection on Carnatic Percussion Instruments*. M.Tech project, Indian Institute of Technology, Department of Electrical Engg., Madras, India.

90. **Kümmerer, M.**, **T. S. Wallis**, and **M. Bethge** (2015). Information-theoretic model comparison unifies saliency metrics. *Proceedings of the National Academy of Sciences*, **112**(52), 16054–16059.

91. **Kümmerer, M.**, **T. S. Wallis**, and **M. Bethge** (2017). Saliency benchmarking: Separating models, maps and metrics. *arXiv preprint arXiv:1704.08615*.

92. **Kuriakose, J.**, **J. C. Kumar**, **P. Sarala**, **H. A. Murthy**, and **U. K. Sivaraman** (2015). Akshara transcription of mrudangam strokes in carnatic music. *In Proceedings of the 21st National Conference on Communications (NCC)*, 1–6.

93. **L. R. Rabiner and R. W. Schafer** (1978). Digital processing of speech signals. *Prentice-Hall, Englewood Cliffs, NJ*.

94. **Lakshmi, A.** and **H. A. Murthy** (2008). A new approach to continuous speech recognition in indian languages. *In Proceedings of National Conference on Communication (NCC)*, 277–281.

95. **Lee, K.-H.**, **S.-I. Kang**, **D.-H. Kim**, and **J.-H. Chang** (2008). A support vector machine-based gender identification using speech signal. *IEICE transactions on communications*, **91**(10), 3326–3329.

96. **Lee, T.-W.** (1998). Independent component analysis. *Springer*, 27–66.

97. **Lee, W.-C.** and **C.-C. Kuo** (2006). Musical onset detection based on adaptive linear prediction. *In Proceedings of IEEE International Conference on Multimedia and Expo, 2006*, 957–960.

98. **Li, J.**, **D. Yu**, **J.-T. Huang**, and **Y. Gong** (2012). Improving wideband speech recognition using mixed-bandwidth training data in cd-dnn-hmm. *In Proceedings of IEEE Spoken Language Technology Workshop (SLT), 2012*, 131–136.

99. **Linsley, D.**, **J. W. Linsley**, **T. Sharma**, **N. Meyers**, and **T. Serre** (2018). Learning to predict action potentials end-to-end from calcium imaging data. *In Proceedings of 52nd Annual Conference on Information Sciences and Systems (CISS), 2018*, 1–6.

100. **Lluís, F.**, **J. Pons**, and **X. Serra** (2018). End-to-end music source separation: is it possible in the waveform domain? *arXiv preprint arXiv:1810.12187*.

101. **Luo, Y.** and **N. Mesgarani** (2018). Tasnet: time-domain audio separation network for real-time, single-channel speech separation. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 696–700.

102. **Madhumurthy, K. V.** and **B. Yegnanarayana** (1989). Effectiveness of representation of signals through group delay functions. *Signal Processing*, **17**, 141–150.

103. **Marchi, E.**, **G. Ferroni**, **F. Eyben**, **L. Gabrielli**, **S. Squartini**, and **B. Schuller** (2014*a*). Multi-resolution linear prediction based features for audio onset detection with bidirectional LSTM neural networks. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*, 2164–2168.

104. **Marchi, E.**, **G. Ferroni**, **F. Eyben**, **S. Squartini**, and **B. Schuller** (2014*b*). Audio onset detection: A wavelet packet based approach with recurrent neural networks. *In Proceedings of International Joint Conference on Neural Networks (IJCNN)*, 3585–3591.

105. **Masri, P.** (1996). *Computer Modeling of Sound for Transformation and Synthesis of Musical Signals*. Ph.D. thesis, University of Bristol, UK.

106. **McAulay, R.** and **T. Quatieri** (1990). Pitch estimation and voicing detection based on a sinusoidal speech model. *In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1990. ICASSP-90*, 249–252 vol.1.

107. **Ming, G.**, **W. Jinfang**, **L. Dongxin**, and **L. Chang** (2013). Depression detection using the derivative features of group delay and delta phase spectrum. *In Proceedings of IEEE International Conference on Advance Computing(IACC), Ghaziabad, India*, 1275–1278.

108. **Muckenhirn, H.**, **M. Magimai-Doss**, and **S. Marcel** (2017). End-to-end convolutional neural network-based voice presentation attack detection. *In Proceedings of IEEE IAPR International Joint Conference on Biometrics (IJCB)*.

109. **Muckenhirn, H.**, **M. Magimai.-Doss**, and **S. Marcel** (2018). Towards directly modeling raw speech signal for speaker verification using CNNs. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2018)*.

110. **Murthy, H. A.** (1991). *Algorithms for Processing Fourier Transform Phase of Signals*. PhD dissertation, Indian Institute of Technology, Department of Computer Science and Engg., Madras, India.

111. **Murthy, H. A.** (1994). Pitch extraction from root cepstrum. *In Proceedings of the 3rd International Conference on Spoken Language Processing,Yokohama, Japan*, 1055–1058.

112. **Murthy, H. A.** and **B. Yegnanarayana** (1991). Formant extraction from group delay function. *speech communication*, **10**(3), 209–221.

113. **Murthy, H. A.** and **B. Yegnanarayana** (2011). Group delay functions and its application to speech processing. *Sadhana*, **36**(5), 745–782.

114. **Murthy, R. M. H. H. A.** and **V. R. R. Gadde** (2007). Significance of joint features derived from the modified group delay function in speech processing. *EURASIP*, **2007**.

115. **Mysore, G. J.**, **P. Smaragdis**, and **B. Raj** (2010). Non-negative hidden markov modeling of audio with application to source separation. *International Conference on Latent Variable Analysis and Signal Separation*, 140–148.

116. **Nagarajan, T.** (2004). *Implicit Systems for Spoken Language Identification*. Ph.D. thesis, Indian Institute of Technology Madras, Department of Computer Science and Engg., Madras, India.

117. **Nagarajan, T.** and **H. A. Murthy** (2006). Language identification using acoustic log-likelihood of syllable-like units. *Journal of Speech Communication*, **48**, pp.913–926.

118. **Nagarajan, T.**, **H. A. Murthy**, and **H. R. M.** (2003). Segmentation of speech into syllable-like units. *In Proceedings of EUROSPEECH, Geneva, Switzerland*, 2893–2896.

119. **Nagarajan, T.**, **V. K. Prasad**, and **H. A. Murthy** (2001). The minimum phase signal derived from the magnitude spectrum and its applications to speech segmentation. *In Proceedings of biennial conference on Signal Processing and Communication (SPCOM-2001)*, 95–101.

120. **Nallapati, R.**, **B. Zhou**, **C. Gulcehre**, **B. Xiang**, *et al.* (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

121. **Noll, A. M.** (1967). Cepstrum pitch determination. *Journal of Acoustical Society of America*, 179–195.

122. **Oppenheim, A. V.** and **R. W. Schafer** (1990). Discrete time signal processing. *Prentice Hall, Inc.,New Jersey*.

123. **Pachitariu, M.**, **C. Stringer**, and **K. D. Harris** (2018). Robustness of spike deconvolution for neuronal calcium imaging. *Journal of Neuroscience*, **38**(37), 7976–7985.

124. **Pachitariu, M.**, **C. Stringer**, **S. Schröder**, **M. Dipoppa**, **L. F. Rossi**, **M. Carandini**, and **K. D. Harris** (2016). Suite2p: beyond 10,000 neurons with standard two-photon microscopy. *Biorxiv*, 061507.

125. **Padmanabahan, R.** (2012). *Studies on Voice Activity Detection and Feature Diversity for speaker recognition*. Ph.D dissertation, Indian Institute of Technology, Department of Computer Science and Engg., Madras, India.

126. **Padmanabhan, R.** and **H. A. Murthy** (2010). Acoustic feature diversity and speaker verification. *In Proceedings of Int. Conf. Spoken Language Processing, Makuhari,Japan*, 2010–2113.

127. **Palaz, D.**, **R. Collobert**, and **M. M. Doss** (2013). Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks. *arXiv preprint arXiv:1304.1018*.

128. **Pedregosa, F.**, **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** (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

129. **Plante, F.**, **G. F. Meyer**, and **W. A. Ainsworth** (1995). A pitch extraction reference database. *In Proceedings of eurospeech-1995*, 837–840.

130. **Pnevmatikakis, E. A.**, **J. Merel**, **A. Pakman**, and **L. Paninski** (2013). Bayesian spike inference from calcium imaging data. *In Proceedings of Asilomar Conference on Signals, Systems and Computers, 2013*, 349–353.

131. **Pnevmatikakis, E. A.**, **D. Soudry**, **Y. Gao**, **T. A. Machado**, **J. Merel**, **D. Pfau**, **T. Reardon**, **Y. Mu**, **C. Lacefield**, **W. Yang**, *et al.* (2016). Simultaneous denoising, deconvolution, and demixing of calcium imaging data. *Neuron*, **89**(2), 285–299.

132. **Pradhan, A.**, **S. Chevireddy**, **K. Veezhinathan**, and **H. A. Murthy** (2010). A low-bit rate segment vocoder using minimum residual energy criteria. *In Proceedings of National Conference on Communication (NCC-2010)*, 246–250.

133. **Qawaqneh, Z.**, **A. A. Mallouh**, and **B. D. Barkana** (2017). Deep neural network framework and transformed mfccs for speaker's age and gender classification. *Knowledge-Based Systems*, **115**, 5–14.

134. **Quatieri, T. F.** (2002). 2-D processing of speech with application to pitch estimation. *In ProceedingS of International Conference in Spoken Language and Processing-2002*.

135. **Rabiner, L.**, **M. Cheng**, **A. Rosenberg**, and **C. McGonegal** (1976). A comparative performance study of several pitch detection algorithms. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, **24**(5), 399–418.

136. **Rajan, P.**, **T. Kinnunen**, **C. Hanilci**, **J. Pohjalainen**, and **P. Alku** (2013). Using group delay functions from all-pole models for speaker recognition. *Proceedings of Int. Conf. Spoken Language Processing, Lyon, France*, 2489–2493.

137. **Rajan, R.** (2017). *Estimation of Pitch in Speech and Music Signals Using Modified Group Delay Functions*. Ph.D. thesis, Indian Institute of Technology Madras.

138. **Rajan, R.** and **H. Murthy** (2013). Group delay based melody monopitch extraction from music. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2013)*, 186–190. ISSN 1520-6149.

139. **Rajan, R.** and **H. A. Murthy** (2016). Modified group delay based multipitch estimation in co-channel speech. *arXiv preprint arXiv:1603.05435*.

140. **Rajesh M. Hegde** (2005). *Fourier transform phase based features for speech recognition*. PhD dissertation, Indian Institute of Technology Madras, Department of Computer Science and Engg., Madras, India.

141. **Raman, C.** (1934). The indian musical drums. *In Proceedings of the Indian Academy of Sciences-Section A*, **1**(3), 179–188.

142. **Ramdinmawii, E.** and **V. Mittal** (2016). Gender identification from speech signal by examining the speech production characteristics. *In Proceedings of International Conference on Signal Processing and Communication (ICSC-2016)*, 244–249.

143. **Ranjan, S.**, **G. Liu**, and **J. H. Hansen** (2015). An i-vector plda based gender identification approach for severely distorted and multilingual darpa rats data. *In Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU-2015),*, 331–337.

144. **Rao, K. S.**, **S. R. M. Prasanna**, and **B. Yegnanarayana** (2007). Determination of instants of significant excitation in speech using Hilbert envelope and group delay function. *IEEE Signal Processing Letters*, **14**(10), 762–765.

145. **Rao, V.** and **P. Rao** (2010). Vocal melody extraction in the presence of pitched accompaniment in polyphonic music. *Audio, Speech, and Language Processing, IEEE Transactions on*, **18**(8), 2145–2154.

146. **Rasipuram, R.**, **R. M. Hegde**, and **H. A. Murthy** (2008*a*). Incorporating acoustic diversity into the linguistic feature space for syllable recognition. *In Proceedings of EUSIPCO-2008*.

147. **Rasipuram, R.**, **R. M. Hegde**, and **H. A. Murthy** (2008*b*). Significance of group delay based acoustic features in the linguistic feature space for syllable recognition. *In Proceedings of Interspeech-2008*.

148. **Rethage, D.**, **J. Pons**, and **X. Serra** (2018). A wavenet for speech denoising. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2018)*, 5069–5073.

149. **R.M.Hegde**, **H. Murthy**, and **G. Rao** (2004). Application of the modified group delay function to speaker identification and discrimination. *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing, Montreal, Canada*, 517–520.

150. **Sainath, T. N.**, **R. J. Weiss**, **K. W. Wilson**, **A. Narayanan**, and **M. Bacchiani** (2016). Factored spatial and spectral multichannel raw waveform cldnns. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016*, 5075–5079.

151. **Salamon, J.** and **E. Gómez** (2012). Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Transactions on Audio, Speech and Language Processing*, **20**, 1759–1770.

152. **Sarada, G. L.**, **L. A.**, **M. H. A.**, and **N. T.** (2009). Automatic transcription of continuous speech into syllable-like units for indian languages. *Sadhana*, **34, Part 2**, pp.221–233.

153. **Schloss, W. A.** (1985). *On the Automatic Transcription of Percussive Music: From Acoustic Signal to High Level Analysis*. Ph.D. thesis, Stanford University, CA, USA. URL `http://ccrma.stanford.edu/STANM/stanms/stanm27/stanm27.pdf`.

154. **Schlüter, J.** and **S. Böck** (2014). Improved Musical Onset Detection with Convolutional Neural Networks. *In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*.

155. **Sebastian, J.**, **M. Kumar**, and **H. A. Murthy** (2016). An analysis of the high resolution property of group delay function with applications to audio signal processing. *Journal of Speech Communication*, 42–53.

156. **Sebastian, J.** and **H. A. Murthy**, Group delay based music source separation using deep recurrent neural networks. *In International Conference on Signal Processing and Communications (SPCOM)*. IEEE, 2016.

157. **Seneff, S.** (1978). Real-time harmonic pitch detector. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **26**(4), 358–365.

158. **Sethu, V.**, **E. Ambikairajah**, and **J. Epps** (2007). Group delay features for emotion detection. *Proceedings of 8th Conference of the International Speech Communication Association, Antwerp, Belgium*, 2273–2276.

159. **Shanmugam, S. A.** and **H. A. Murthy** (2014*a*). Group delay based phone segmentation for HTS. *National Conference on Communications 2014 (NCC-2014), Kanpur, India*.

160. **Shanmugam, S. A.** and **H. A. Murthy** (2014*b*). A hybrid approach to segmentation of speech using group delay processing and HMM based embedded reestimation. *in Proceedings of Fifteenth Annual Conference of the International Speech Communication Association*.

161. **Shi, G.**, **M. Shanechi**, and **P. Aarabi** (2006). On the importance of phase in human speech recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, **14**(5), 1867 –1874.

162. **Siddharthan, R.**, **P. Chatterjee**, and **V. Tripathi**, A study of harmonic overtones produced in indian drums. *In Physics Education*. 1994.

163. **Simpson, A. J.** (2015). Probabilistic binary-mask cocktail-party source separation in a convolutional deep neural network. *arXiv preprint arXiv:1503.06962*.

164. **Smits, R.** and **B. Yegnanarayana** (1995). Determination of instants of significant excitation in speech using group delay function. *Speech and Audio Processing, IEEE Transactions on*, **3**(5), 325–333. ISSN 1063-6676.

165. **Sondhi, M. M.** (1968). New methods of pitch extraction. *Audio and Electroacoustics, IEEE Transactions on*, **16**(2), 262–266.

166. **Speiser, A.**, **J. Yan**, **E. W. Archer**, **L. Buesing**, **S. C. Turaga**, and **J. H. Macke** (2017). Fast amortized inference of neural activity from calcium imaging data with variational autoencoders, 4024–4034.

167. **Sri Rama Murty, K.** and **B. Yegnanarayana** (2008). Epoch extraction from speech signals. *IEEE Transactions on Audio,Speech and Language Processing*, **16**(8), 1602–1613.

168. **Stosiek, C.**, **O. Garaschuk**, **K. Holthoff**, and **A. Konnerth** (2003). In vivo two-photon calcium imaging of neuronal networks. *Proceedings of the National Academy of Sciences*, **100**(12), 7319–7324.

169. **Sutskever, I.**, **O. Vinyals**, and **Q. V. Le** (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 3104–3112.

170. **Svoboda, H.** (2015). Simultaneous imaging and loose-seal cell-attached electrical recordings from neurons expressing a variety of genetically encoded calcium indicators. *GENIE project, Janelia Farm Campus, CRCNS. org*.

171. **Talkin, D.** (1995). A robust algorithm for pitch tracking (RAPT). *Speech Coding and Synthesis, Elsevier*.

172. **Theis, L.**, **P. Berens**, **E. Froudarakis**, **J. Reimer**, **M. Román Rosón**, **T. Baden**, **T. Euler**, **A. S. Tolias**, and **M. Bethge** (2016). Benchmarking Spike Rate Inference in Population Calcium Imaging. *Neuron*, **90**(3), 471–82. ISSN 1097-4199. URL `http://dx.doi.org/10.1016/j.neuron.2016.04.014http://www.cell.com/article/S0896627316300733/fulltext`.

173. **Theis, L.**, **A. M. Chagas**, **D. Arnstein**, **C. Schwarz**, and **M. Bethge** (2013). Beyond GLMs: A Generative Mixture Modeling Approach to Neural System Identification. *PLoS Computational Biology*, **9**(11), e1003356. ISSN 1553734X.

174. **Thiemann, J.**, **N. Ito**, and **E. Vincent** (2013). The diverse environments multi-channel acoustic noise database (DEMAND): A database of multi-channel environmental noise recordings. *Proceedings of Meetings on Acoustics ICA2013*, **19**(1), 035081.

175. **Thiruvaran, T.**, **E. Ambikairajah**, and **J. Epps** (2007). Group delay features for speaker recognition. *In Proceedings of 6th International Conference on Information, Communications and Signal Processing, Singapore*.

176. **Tian, M.**, **A. Srinivasamurthy**, **M. Sandler**, and **X. Serra** (2014). A study of instrument-wise onset detection in beijing opera percussion ensembles. *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014*, 2159–2163.

177. **Trigeorgis, G.**, **F. Ringeval**, **R. Brueckner**, **E. Marchi**, **M. A. Nicolaou**, **B. Schuller**, and **S. Zafeiriou**, Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. *In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016.

178. **Venkataramani, S.**, **J. Casebeer**, and **P. Smaragdis** (2017). Adaptive front-ends for end-to-end source separation. *In Proceedings of NIPS*.

179. **Victor, J. D.** and **K. P. Purpura** (1996). Nature and precision of temporal coding in visual cortex: a metric-space analysis. *Journal of neurophysiology*, **76**(2), 1310–1326.

180. **Vincent, E.**, **R. Gribonval**, and **C. Févotte** (2006). Performance measurement in blind audio source separation. *Audio, Speech, and Language Processing, IEEE Transactions on*, **14**(4), 1462–1469.

181. **Vinodh, M.**, **A. Bellur**, **K. B. Narayan**, **M. D. Thakare**, **A. Susan**, **N. Suthakar**, and **H. A. Murthy** (2010). Using polysyllabic units for text to speech synthesis in indian languages. *In Proceedings of National Conference on Communication, Indian Institute of Technology, Chennai*.

182. **Virtanen, T.** (2003). Algorithm for the separation of harmonic sounds with time-frequency smoothness constraint. *In Proceedings of Int. Conf. on Digital Audio Effects (DAFx)*, 35–40.

183. **Vogelstein, J. T.**, **A. M. Packer**, **T. A. Machado**, **T. Sippy**, **B. Babadi**, **R. Yuste**, and **L. Paninski** (2010). Fast nonnegative deconvolution for spike train inference from population calcium imaging. *Journal of neurophysiology*, **104**(6), 3691–3704.

184. **Vogelstein, J. T.**, **B. O. Watson**, **A. M. Packer**, **R. Yuste**, **B. Jedynak**, and **L. Paninski** (2009). Spike inference from calcium imaging using sequential monte carlo methods. *Biophysical journal*, **97**(2), 636–655.

185. **Wang, T. T.** and **T. F. Quatieri** (2009). 2-D processing of speech for multi-pitch analysis. *In Proceedings of Int. Conf. Spoken Language Processing,Brighton, United Kingdom*.

186. **Weiss, R. J.** and **D. P. Ellis** (2006). Estimating single-channel source separation masks: Relevance vector machine classifiers vs. pitch-based masking. *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition: SAPA2006: 16 September 2006, Pittsburgh, PA*, 31–36.

187. **Weninger, F.**, **J. R. Hershey**, **J. Le Roux**, and **B. Schuller** (2014). Discriminatively trained recurrent neural networks for single-channel speech separation. *In Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP), 2014*, 577–581.

188. **Wilt, B. A.**, **J. E. Fitzgerald**, and **M. J. Schnitzer** (2013). Photon shot noise limits on optical detection of neuronal spikes and estimation of spike timing. *Biophysical journal*, **104**(1), 51–62.

189. **W.S-URL** (2012). http://www.speech.kth.se/wavesurfer/. *URL*.

190. **Wu, K.** and **D. G. Childers** (1991). Gender recognition from speech. part I: Coarse analysis. *Journal of the Acoustical society of America*, **90**(4), 1828–1840.

191. **Xu, Y.**, **J. Du**, **L.-R. Dai**, and **C.-H. Lee** (2014). An experimental study on speech enhancement based on deep neural networks. *IEEE Signal processing letters*, **21**(1), 65–68.

192. **Xu, Yong**, **Du, Jun**, **Dai, Li-Rong**, and **Lee, Chin-Hui** (2015). A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, **23**(1), 7–19.

193. **Yaksi, E.** and **R. W. Friedrich** (2006). Reconstruction of firing rate changes across neuronal populations by temporally deconvolved ca 2+ imaging. *Nature methods*, **3**(5), 377.

194. **Yegnanarayana, B.** (1979). Formant extraction from linear prediction phase spectra. *Journal of Acoustical Society of America*, **63**, 1638–1640.

195. **Yegnanarayana, B.** and **H. A. Murthy** (1992). Significance of group delay functions in spectrum estimation. *IEEE Trans. Signal Processing*, **40**(9), 2281–2289.

196. **Yegnanarayana, B.**, **H. A. Murthy**, and **V. R. Ramachandran** (1991). Processing of noisy speech using modified group delay functions. *ICASSP*, pp.945–948.

197. **Yegnanarayana, B.**, **S. Rajendran**, **V. R.Ramachandran**, and **A. S. Madhukumar** (1994). Significance of knowledge sources for a text-to-speech system for Indian languages. *Sadhana, pp.147-169*.

198. **Yegnanarayana, B.**, **D. K. Saikia**, and **T. R. Krishnan** (1984). Significance of group delay functions in signal reconstruction from spectral magnitude or phase. *IEEE Trans. Acoustics Speech and Signal Processing*, **ASSP-32**(3), 610–623.

199. **Yip, P.** and **K. R. Rao** (1997). Discrete cosine transform: Algorithms, advantages and applicatons. *Academic Press, USA*.

200. **Zeng, Y.-M.**, **Z.-Y. Wu**, **T. Falk**, and **W.-Y. Chan** (2006). Robust gmm based gender classification using pitch and rasta-plp parameters of speech. *In Proceedings of International Conference on Machine Learning and Cybernetics, 2006*, 3376–3379.

LIST OF PUBLICATIONS

Journal Articles

1. Jilt Sebastian, Mari Ganesh Kumar, Venkata Subramanian Viraraghavan, Mriganka Sur and Hema A. Murthy *"Spike Estimation From Fluorescence Signals Using High-Resolution Property of Group Delay"*, IEEE transactions on signal processing. Volume 67, Issue 11, (2019), DOI:10.1109/TSP.2019.2908913.

2. Jilt Sebastian, Manoj Kumar P. A. and Hema A. Murthy, Analysis of High Resolution Property of Group Delay function with Applications to Audio Signal Processing, *Journal of Speech Communication, Elsevier*, 81, pages: 42-53, (2016), URL: https://doi.org/10.1016/j.specom.2015.12.008.

Conference Papers

1. Jilt Sebastian, Manoj Kumar P. A., D. S. Pavankumar, Mathew Magimai.-Doss, Hema A. Murthy and Shrikanth Narayanan, *"Denoising and Raw-waveform Networks for Weakly-Supervised Gender Identification on Noisy Speech"*, Interspeech 2018, Hyderabad, India, URL: http://dx.doi.org/10.21437/Interspeech.2018-2321.

2. Bogdan Vlasenko, Jilt Sebastian, D. S. Pavankumar and Mathew Magimai.-Doss, *"Implementing Fusion Techniques for the Classification of Paralinguistic Information"*,Interspeech 2018, Hyderabad, India,
URL: http://dx.doi.org/10.21437/Interspeech.2018-2360.

3. Jilt Sebastian and Hema A. Murthy, *"Onset Detection in Composition Items of Carnatic Music"*, ISMIR 2017, Suzhou, China,
URL: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/91_Paper.pdf.

4. Jilt Sebastian, Mari Ganesh Kumar M, Y. S. Sreekar, Rajeev Vijay Rikhye, Mriganka Sur and Hema A. Murthy, *"GDspike: An Accurate Spike estimation Algorithm from Noisy Calcium Flourescence Signals"*, ICASSP 2017, New Orleans, United States, URL: 10.1109/ICASSP.2017.7952315.

5. Jilt Sebastian and Hema A. Murthy, *"Group Delay Based Music Source Separation Using Deep Recurrent Neural Networks"*, SPCOM 2016, Bangalore, India, URL: 10.1109/SPCOM.2016.7746672.

6. Jilt Sebastian, Manoj Kumar P. A. and Hema A. Murthy, *"Pitch Estimation From Speech Using Grating Compression Transform on Modified Group-Delay-gram"*, NCC 2015, Mumbai, India, URL: 10.1109/NCC.2015.7084899.

7. Manoj Kumar P. A., Jilt Sebastian and Hema A. Murthy, *"Musical Onset Detection on Carnatic Percussion Instruments"*, NCC 2015, Mumbai, India, URL: 10.1109/NCC.2015.7084897.

DOCTORAL COMMITTEE

Chairperson: Prof. V. Kamakoti (HoD's Nominee)
Department of Computer Science and Engineering
Indian Institute of Technology Madras

Research Advisor: Prof. Hema A. Murthy
Department of Computer Science and Engineering
Indian Institute of Technology Madras

Members: Prof. C. Chandra Sekhar
Department of Computer Science and Engineering
Indian Institute of Technology Madras

Prof. R. Aravind
Department of Computer Science and Engineering
Indian Institute of Technology Madras

Dr. Sutanu Chakraborti
Associate Professor
Department of Electrical Engineering
Indian Institute of Technology Madras

CURRICULUM VITAE

1. NAME : Jilt Sebastian

2. DATE OF BIRTH : August 17^{th}, 1989

3. PERMANENT ADDRESS : Chittappanattu House,
Kanchiyar P.O,
Labbakkada - 685 511,
Idukki, Kerala
Email: jiltsebastian@gmail.com
Phone: +91-9962876798

4. EDUCATIONAL QUALIFICATIONS

Bachelor of Technology (B.Tech.)

Year of Completion	:	2011
Institution	:	Amal Jyothi College of Engineering, Kottayam
Specialization	:	Electronics & Communication Engineering