Estimation of Pitch in Speech and Music Signals Using Modified Group Delay Functions

A THESIS

submitted by

RAJEEV RAJAN

for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.
MARCH 2017
This is to certify that the thesis titled, “Estimation of Pitch in Speech and Music Signals Using Modified Group Delay Functions”, submitted by Rajeev Rajan, 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:
ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my guide Prof. Hema A. Murthy for her constant support and guidance during the entire period of my research. She spent enough time for discussions to teach me speech and music processing. She was always helpful and a source of constant encouragement throughout my studies.

I would like to thank Heads of the department during my research period Prof. C. Siv Ram Murthy, Prof. P. Sreenivasa Kumar, Prof. Krishna Moorthy Sivalingam for providing excellent lab facilities to carry out the research work in the department.

I am thankful to my Doctoral Committee members Dr. D. Janakiram, Dr. Deepak Khemani, Dr. Rajagopalan A.N, Dr. Sutanu Chakroborthi and Dr.Venktesh Ramaiyan for their valuable suggestions and feedback during the progress review meetings. I take this opportunity to thank Dr. C. Chandrashekhari for his timely advice and suggestions during various presentations in speech lab meetings and seminars. I would like to thank AICTE and Directorate of Technical Education, Government of Kerala for giving me an opportunity to study in IIT Madras.

All the members (past and present) of the DON Lab and MS Lab, were helpful whenever I need assistance during my studies in the campus. I thank all my colleagues in the lab, with whom I have interacted at some point: Padmanabhan, Abhijith, Golda, Sarala, Srikant, Subhadeep, Saad, Vignesh, Ashwin, Shrey Dutta, Akshay, Cyriac, Raghav, Manaswi, Jilt, Jom, Mari, Krishnaraj, Karthik, Saranya, Anusha, Jeena, Arun, Anju, Manish, Nauman, Hari, Kasthuri, Gayathri and Mohana. I would like to thank my friends in the institute, Sangeetha Jose, Preethi madam, Subhashini, Shajimohan, Ajeesh Ramanujam, Jayesh, Manoj, Akes, Rahul, Lio, Joe and Sam Sreenivas for their constant support and guidance.

I also thank all the staff members in the Department of computer science and engineering, IIT Madras and QIP office for their assistance and cooperation. I express my gratitude to my wife, Beena for her encouragement and understanding throughout my stay at IIT Madras. With great pleasure, I would like to give special thanks to my parents, sister, father-, mother-, sisters- and brothers-in-law, for their support, prayers,
patience and encouragement throughout my studies. I express my deep sense of love to my kids, Nanma and Nathu. I thank GOD almighty for all the blessings, He has bestowed upon me in my life.

Rajeev Rajan
ABSTRACT

KEYWORDS: melody, modified group delay, cepstral smoothing, multipitch, source-group delay, genre, power spectrum, tracking

The estimation of fundamental frequency, or the pitch of constituent components of audio signals has a wide range of applications in computational auditory scene analysis (CASA), prosody analysis, source separation, speaker identification and music signal processing. This thesis addresses pitch estimation in mixed speech and predominant pitch estimation in polyphonic music. In the first task, predominant melodic pitch is extracted from music using modified group delay functions. Predominant pitch extraction is performed using two different flavours of the modified group delay function. In the first approach, modified group delay analysis is performed directly on the music signal. Choice of the window for cepstral smoothing, and pitch range play a crucial role in the accurate estimation of pitch. To overcome the stringent pitch range constraint in the scheme, system characteristics are annihilated from the spectrum in the second approach, and the flattened spectrum is processed using modified group delay functions to obtain an estimate of pitch. Multi-resolution analysis is employed to capture dynamic variation of melody and dynamic programming is incorporated to ensure consistency in pitch tracking. Automatic music genre classification using features computed from the melodic contour is also explored.

In the second task, individual pitch tracks are estimated from co-channel speech using modified group delay functions. When utterances of different speakers are combined additively, and transmitted through a single channel, pitch cues of individual sources will be weakened by the presence of mutual interference. In the proposed work, flattened spectrum is analysed for predominant pitch in the first pass. Later, the estimated pitch and its harmonics are annihilated from the spectrum, and the residual spectrum is once again processed to obtain the next prominent pitch in the signal. The individual pitch trajectories are refined in pitch grouping and post processing stage.

In the third task, a variant of source-group delay representation is introduced. The
flattened power spectrum is analyzed using group delay algorithm, followed by discrete cosine transform to convert the source based modified group delay function to meaningful features. The novel feature is effectively used for two applications, namely audio indexing and estimation of number of speakers in mixed speech.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS .............................................. i

ABSTRACT ....................................................... iii

LIST OF TABLES ............................................... viii

LIST OF FIGURES ................................................ xi

ABBREVIATIONS ............................................... xii

NOTATION .......................................................... xiii

1 INTRODUCTION .................................................. 1
  1.1 Motivation ................................................ 1
  1.2 Objective of the thesis .................................. 2
  1.3 Organization of the thesis .............................. 2
  1.4 Major contributions of the thesis ..................... 3

2 Theory of Group Delay and Modified Group Delay Functions in Speech Processing ........................................... 4
  2.1 Introduction .............................................. 4
  2.2 Group delay in speech signal processing ............. 4
    2.2.1 Properties of group delay functions .............. 7
  2.3 Modified group delay function (MODGD) ............. 12
  2.4 Modified group delay on flattened spectrum
    MODGD (Source) ............................................ 15
  2.5 Group delay applications in speech and music ....... 17
    2.5.1 Pitch and formant estimation .................... 18
    2.5.2 Speech and speaker recognition ................. 19
    2.5.3 Speech synthesis ................................... 20
    2.5.4 Speech emotion recognition ..................... 21
    2.5.5 Voice activity detection ......................... 21
    2.5.6 Chirp group delay processing and ASR .......... 22
    2.5.7 Tonic identification ............................. 22
    2.5.8 Onset detection ................................... 23

3 Melody Extraction in Polyphonic Music .................. 24
3.1 Introduction .................................................. 24
3.2 Music information retrieval (MIR) .......................... 24
3.3 Melodic pitch estimation ................................. 26
3.4 Melody extraction approaches ............................ 29
  3.4.1 Salience based approaches ......................... 30
  3.4.2 Source separation based approaches .............. 30
  3.4.3 Salience and Source separation based approaches 31
  3.4.4 Data driven approaches ............................. 32
  3.4.5 Acoustic/Musicological model based approaches 32
3.5 Evaluation methodology ................................. 33
3.6 Melody extraction: Challenges and Applications .......... 34
3.7 Audio melody extraction from music using modified group delay functions ................................ 37
3.8 Melodic-pitch estimation using MODGD (Direct) ........ 38
  3.8.1 Theory of predominant melodic pitch estimation 38
  3.8.2 Performance evaluation .............................. 40
3.9 Melody pitch extraction using flattened spectrum .......... 43
  3.9.1 Monopitch estimation using MODGD (Source) .... 43
  3.9.2 Melody pitch extraction using MODGD (Source) 44
  3.9.3 Transient analysis using a Multi-resolution Framework 47
  3.9.4 Pitch tracking using dynamic programming .......... 48
  3.9.5 Voice Activity Detection (VAD) ..................... 51
  3.9.6 Performance Evaluation ............................. 52
  3.9.7 Results, analysis and discussion .................. 53
3.10 Automatic music genre classification ............... 58
  3.10.1 Proposed method .................................. 59
  3.10.2 Modified group delay feature (MODGDF) ........ 60
  3.10.3 Melodic Features ................................. 60
  3.10.4 Evaluation Dataset ............................... 63
  3.10.5 Results and analysis .............................. 64

4 Two-pitch Tracking in Co-channel Speech .............. 66
  4.1 Introduction ............................................ 66
  4.2 Multipitch Estimation .................................... 66
  4.3 Multipitch estimation algorithms : State-of-the-art approaches .............................. 68
  4.4 Evaluation methodology ............................... 71
  4.5 Applications of multipitch estimation .......... 72
  4.6 Modified group delay based two-pitch tracking in co-channel speech .............................. 73
    4.6.1 Theory of pitch detection using modified group delay functions 74
    4.6.2 Proposed system description ...................... 75
## LIST OF TABLES

2.1 Algorithm for the computation of MODGDF ........................................... 13

3.1 Music information retrieval and level of specificity ............................... 26
3.2 Comparison of $\sigma_e$ and RPA .............................................................. 42
3.3 Comparison of RPA and $\sigma_e$ for MIR-1K dataset ................................. 42
3.4 Description of the datasets used for the evaluation ............................... 52
3.5 Comparison of results for ADC-2004 submitted in 2011/2012/2013/2014 evaluation ................................................................. 56
3.6 Comparison of results for MIREX-2008 submitted in 2011/2012/2013/2014 evaluation ................................................................. 56
3.7 Comparison of $\sigma_e$, RPA, RCA for LabROSA training dataset .......... 56
3.8 Comparison of $\sigma_e$, RPA, RCA for Carnatic music dataset .............. 57
3.9 Overall accuracy .................................................................................. 63
3.10 Classification accuracies obtained by existing approaches ............... 63
3.11 Confusion matrix of fusion (MODGDF + melodic) experiment ........ 65

4.1 Multipitch estimation-Summarization of few algorithms ....................... 71
4.2 Category of mixtures for Dataset:1 and 2 ............................................. 85
4.3 Performance evaluation on Dataset-1 .................................................. 86
4.4 Performance evaluation on Dataset-2 .................................................. 86
4.5 Comparison of $E_{fs}$ corresponding to $\text{Accuracy}_{10}$ (in Hz) (Dataset-1) ................................................................. 88
4.6 Comparison of $E_{fs}$ corresponding to $\text{Accuracy}_{10}$ (in Hz) (Dataset-2) ................................................................. 88
4.7 Comparison of accuracy for different TMR : Database:1 ................. 89
4.8 Comparison of accuracy for different TMR : Database:2 ................. 89
4.9 Pitch estimation accuracy for three speaker mixed speech .............. 91

5.1 Confusion matrix for number of speaker estimation task. SP-1, SP-2 and SP-3 denote one speaker case, two speaker case and three speaker case, respectively. Class wise accuracy is given as the last column entry. ................................................................. 96
5.2 Confusion matrix for audio indexing task. Class wise accuracy is given as the last column entry. ................................................................. 98
5.3 Results on Speech/Music detection on audio recordings. Correctly indexed speech/music frames in each recording is entered in %. Global accuracy is given as the entry in the last column. ................................................................. 99
5.4 Performance comparison of Speech/Music detection in database, DB2 101
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>(a) Fourier transform phase, (b) Unwrapped phase, (c) Group delay.</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>Illustration of additive property of the group delay functions. Three signals $x[n]$, $h[n]$ and $y[n]$, where $y[n] = x[n] * h[n]$ (upper pane); magnitude spectra of three signals (middle pane); group delay functions of three signals (the lower pane).</td>
<td>7</td>
</tr>
<tr>
<td>2.3</td>
<td>Illustrating the high-resolution property for a two conjugate pole system. The faster decay of group delay in comparison to magnitude spectrum can be seen.</td>
<td>9</td>
</tr>
<tr>
<td>2.4</td>
<td>(a) Noisy composite signal, (b) Magnitude Spectrum, (c) Group delay spectrum of (a), (d) Modified group delay spectrum of (a).</td>
<td>9</td>
</tr>
<tr>
<td>2.5</td>
<td>(a) $z$-plane with two conjugate pole pairs, (b) magnitude spectrum of the system shown in (a), (c) group delay spectrum of the system shown in (a).</td>
<td>11</td>
</tr>
<tr>
<td>2.6</td>
<td>(a) The $z$-plane with two conjugate pole pairs, (b) The group delay spectrum of the system shown in (a), (c) The $z$-plane with two pole pairs and zeros added, (d) the group delay spectrum of the system shown in (c), (e) The $z$-plane with zeros pushed radially inward, (f) the group delay spectrum of the system shown in (e).</td>
<td>13</td>
</tr>
<tr>
<td>2.7</td>
<td>(a) Frame of synthetic signal, (b) Magnitude spectrum, (c) Smeothened spectrum, (d) Flattened spectrum.</td>
<td>16</td>
</tr>
<tr>
<td>2.8</td>
<td>Cepstrum method of smoothing</td>
<td>16</td>
</tr>
<tr>
<td>2.9</td>
<td>Root cepstrum method</td>
<td>17</td>
</tr>
<tr>
<td>2.10</td>
<td>(a) A speech utterance, (b) First formant frequency estimated using MODGD, (c) Pitch estimated using MODGD (Wavesurfer reference is also given).</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>Music information retrieval system</td>
<td>25</td>
</tr>
<tr>
<td>3.2</td>
<td>Melody extracted for a song</td>
<td>27</td>
</tr>
<tr>
<td>3.3</td>
<td>Prominent melodic pitch (grey contour) on spectrogram.</td>
<td>27</td>
</tr>
<tr>
<td>3.4</td>
<td>Musical notation and its metadata representation for the song <em>Happy Birthday to You</em> (Source: <a href="https://en.wikipedia.org/wiki/HappyBirthdaytoYou">https://en.wikipedia.org/wiki/HappyBirthdaytoYou</a>)</td>
<td>28</td>
</tr>
<tr>
<td>3.5</td>
<td>(a) Time domain signal, (b) melodic contour (c) spectrogram</td>
<td>29</td>
</tr>
<tr>
<td>3.6</td>
<td>Melodic contours with different degrees of polyphony. (a) low, (b) medium, and (c) high. (Melody is extracted using pYIN algorithm)</td>
<td>35</td>
</tr>
<tr>
<td>3.7</td>
<td>Octave errors in melodic pitch estimation</td>
<td>36</td>
</tr>
<tr>
<td>3.8</td>
<td>Two instances of melodic motif of rāga Kāmboji in an ālāpana</td>
<td>37</td>
</tr>
<tr>
<td>3.9</td>
<td>(a) Cepstral window for cepstral smoothing, (b) Peaks correspond to pitch and formants in MODGD plot.</td>
<td>39</td>
</tr>
</tbody>
</table>
3.10 MODGD plot for a speech frame. Stems indicate formant locations and pitch. Wavesurfer (W.S) and estimated (Est) values are shown in boxes.

3.11 (a) MODGD plot for a frame of a music sample, (b) Melody pitch extracted using MODGD (Direct).

3.12 (a) Frame of a speech, (b) Flattened power spectrum, (c) Peaks in the MODGD feature space, (d) Pitch estimated for the entire utterance with reference.

3.13 Overview of MODGD (Source) based melody pitch extraction.

3.14 (a) A frame of music signal, (b) Magnitude spectrum of (a), (c) Spectral envelope of (a), (d) Flattened spectrum of (a).

3.15 Computation of optimal path by dynamic programming.

3.16 (a) MODGD plot for a frame, (b) Melody pitch extraction for opera fem4.wav using MODGD (Source).

3.17 (a) Melody pitch extracted for North Indian classical audio segment, (b) Melody pitch extracted for a Carnatic music segment.

3.18 Raw pitch accuracy of ADC-2004 dataset in box plot.

3.19 Results on adaptive windowing on three window sizes.

3.20 (a) RPA and RCA for DB1 [less accompaniments], (b) RPA and RCA for DB2 [5-6 accompaniments], (W.S : Wavesurfer, pYIN : Probabilistic YIN, MEL : MELODIA, MODGD : MODGD (Source)).

3.21 Block diagram of the proposed system.

3.22 Spectrogram (top) and Modgdgram (bottom) for a song from jazz genre.

3.23 Kernel density estimate of melodic pitch for two genres. (a) Pop, (b) Jazz.

3.24 Genre-wise classification accuracy for all the systems.

4.1 Monopitch estimation algorithms fail at mixed speech example. Wavesurfer pitch is plotted for a two-speaker mixed speech. Individual reference pitches are also shown.

4.2 Iterative multipitch estimation (Klapuri, 2008).

4.3 Block diagram of the proposed method. Dotted block represents the MODGD (Source) based predominant pitch estimation algorithm.

4.4 Notch filter magnitude response and pole-zero plot.

4.5 (a) A frame of synthetic mixture, (b) Flattened spectrum of frame in (a), (c) MODGD plot in the first pass. (d) MODGD of (a) after the application of comb filter (second pass), (e) Pitch extracted for the mixed synthetic speech segment.

4.6 (a) MODGD plot for a frame in first pass and residual spectrum (dotted line) for a real speech mixture, (b) Presence of stray values in the pitch contour are marked in circle.

4.7 (a) Harmonic energy of a segment, (b) Initial pitch estimates of the segment in (a), (c) Final pitch trajectory in the post processing stage.
4.8 (a) Initial pitch estimates for a speech mixture, (b) Final pitch trajectories estimated using the proposed algorithm, (c) Pitch trajectories estimated using WWB algorithm. ............................................. 83
4.9 Room impulse response .................................................. 85
4.10 Speakers with two closed pitch trajectories ......................... 89
4.11 (a) Modified group delay function computed for a speech frame, (b) Pitch contours computed for the speech segment. ............... 90
4.12 Pitch contours computed for a natural speech mixture with three speakers. .................................................. 90

5.1 (a) MODGD on flattened power spectrum of music frame, (b) MODGD on flattened power spectrum of a speech frame .............. 94
5.2 Two dimensional visualization of speech-music data with SMCF feature using Sammon mapping ................................. 98
5.3 Performance evaluation of an audio sample, 1 stands for speech and 2 stands for music ................................. 100

B.1 Selection of plug-in on Sonic Visualiser ............................... 108
B.2 Melodic sequences of proposed algorithm along with MELODIA reference. ............................................. 109
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC</td>
<td>Audio Description Contest</td>
</tr>
<tr>
<td>AMDF</td>
<td>Average Magnitude Difference Function</td>
</tr>
<tr>
<td>ASR</td>
<td>Automatic Speech Recognition</td>
</tr>
<tr>
<td>CASA</td>
<td>Computational Auditory Scene Analysis</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Models</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Model</td>
</tr>
<tr>
<td>HPSS</td>
<td>Harmonic and Percussive Sound Separation</td>
</tr>
<tr>
<td>LPC</td>
<td>Linear Predictive Coefficients</td>
</tr>
<tr>
<td>MFCC</td>
<td>Mel-Frequency Cepstral Coefficients</td>
</tr>
<tr>
<td>MFDP</td>
<td>Mel-Frequency Delta-Phase</td>
</tr>
<tr>
<td>MIR</td>
<td>Music Information Retrieval</td>
</tr>
<tr>
<td>MODGD</td>
<td>Modified Group Delay</td>
</tr>
<tr>
<td>MODGDF</td>
<td>Modified Group Delay Feature</td>
</tr>
<tr>
<td>NCCF</td>
<td>Normalized Cross Correlation Functions</td>
</tr>
<tr>
<td>NMF</td>
<td>Non-negative Matrix Factorization</td>
</tr>
<tr>
<td>pYIN</td>
<td>Probabilistic YIN</td>
</tr>
<tr>
<td>QBSH</td>
<td>Query By Humming/Singing</td>
</tr>
<tr>
<td>RIR</td>
<td>Room Impulse Response</td>
</tr>
<tr>
<td>SACF</td>
<td>Summary Auto-Correlation Function</td>
</tr>
<tr>
<td>TMR</td>
<td>Target-to-Masker Ratio</td>
</tr>
<tr>
<td>VAD</td>
<td>Voice Activity Detection</td>
</tr>
</tbody>
</table>
NOTATION

\(\sigma_e\) Standard deviation of pitch detection
\(\alpha, \gamma\) Modified group delay parameters
\(\omega\) Angular frequency
\(\tau_p\) Group delay derived from Fourier transform phase
\(\tau\) Group delay
\(\tau_m\) Modified group delay
\(f_0\) Fundamental frequency
\(\rho\) Autocorrelation coefficient
\(z\) \(z\)-transform
\(\gamma\) root-cepstrum parameter
\(p_s\) Reference pitch
\(p_p\) Estimated pitch
\(e\) Mean of pitch error
\(slen\) Signal length
\(tlen, hlen\) Hamming window parameters
\(\delta\) Cepstral smoothing -smoothing parameter
\(\sigma\) Cepstral smoothing-roll of parameter
\(P_{\text{min}}\) Minimum pitch
\(P_{\text{max}}\) Maximum pitch
\(F(c)\) Value of pitch salience peak at \(c\).
\(F_{\text{max}}\) Maximum value of pitch salience
\(c_t\) Transition cost
\(c_l\) Local cost
\(F_r(l)\) Reference frequency
\(F_e(l)\) Estimated frequency
\(\mu_p\) Mean pitch height
CHAPTER 1

INTRODUCTION

Fundamental frequency \((f_o)\) estimation continues to be a topical area of research in speech and music technology domains. The fundamental frequency is usually the lowest frequency component or partial present in the signal. Research in this area goes under the name of pitch detection. Choosing an \(f_o\) estimator for a speech/song discrimination is a difficult task because detectors that work well for music may give degraded performance in speech and vice versa (Gerhard, 2003). In this thesis, mainly three tasks are addressed using group delay analysis. In the first task, predominant melodic pitch is estimated in polyphonic music using modified group delay functions. Two schemes, based on modified group delay functions are proposed to extract predominant melodic pitch sequence from polyphonic music. Secondly, an algorithm for two-pitch tracking in co-channel speech is proposed and performed on a number of speech mixtures. Thirdly, a variant of group delay feature is derived and is effectively used for estimating number of speakers in mixed speech and speech/music detection.

1.1 Motivation

From the multicultural and multidisciplinary aspects of music, it is obvious that the challenges facing music information retrieval (MIR) research and development are far from trivial (Downie, 2004). Many difficulties still remain to be overcome before the content-based MIR systems become reality. If it is query for a song or karaoke separation from an orchestration, the user expects an efficient system naturally. Numerous techniques are available to retrieve useful information from music. Group delay function, which is defined as the negative derivative of unwrapped Fourier transform phase is the basis of the proposed tasks discussed in the thesis. The importance of phase in speech perception and recognition has been already established (Murthy and Yegnanarayana, 2011; Paliwal, 2003). The process of phase unwrapping; recovering the original phase values from the principal values is a challenging task. The advantage of group delay is that it can be computed directly from the signal. The competitive
performance of modified group delay function in speech was the key motivation to apply the same in predominant melodic pitch extraction in polyphonic music. Computational methods replaced ‘searching and retrieval’ of songs based on “Meta-data” from digital libraries. Automatic melody extraction is an inevitable step for query by humming/singing (QBSH), music score following and computational music analysis. The proposed method opens the door to a wide range of music processing applications.

1.2 Objective of the thesis

The objective is to study the usefulness of modified group delay function for pitch extraction in speech and predominant melody extraction in music.

1.3 Organization of the thesis

- Chapter 2 reviews the theory of group delay functions and modified group delay functions. Properties of group delay functions are also listed with suitable examples. Few applications of modified group delay functions in speech and music signal processing are also discussed.

- Chapter 3 discusses the approaches, evaluation strategies in melody extraction task followed by the proposed schemes. Two algorithms are described for predominant melodic pitch estimation. In the first approach, group delay analysis is performed directly on music signal. In the second approach, the speech/music signal is preprocessed to remove the effect of the vocal tract. This residual signal (in the frequency domain) is then subjected to modified group delay analysis to estimate the predominant pitch. The performance is evaluated using the standard MIREX framework. Automatic music genre classification using fusion of high level melodic features and low level features is also discussed.

- Chapter 4 discusses the proposed algorithm for pitch estimation in a two speaker environment. The modified group delay based pitch extraction algorithm is used to estimate the predominant pitch. Next, the predominant pitch and its harmonics are eliminated using a comb filter. The residual signal is then subjected to group delay analysis to estimate the pitch of the second speaker.

- Chapter 5 discusses the feature derived from the source group delay and its applications. The modified group delay feature is used in two speech and music related tasks, namely 1) to determine the number of speakers in a speech mixture, and 2) to distinguish between speech and music in a monophonic recording.
1.4 Major contributions of the thesis

The major contributions of the thesis are:

- Two variants of modified group delay function based pitch extraction are proposed.
- Modified group delay function based music genre classification is proposed.
- Two-pitch tracking in mixed speech is addressed using modified group delay functions.
- Speech/music detection in a mixed audio recording using modified group delay functions is performed.
CHAPTER 2

Theory of Group Delay and Modified Group Delay Functions in Speech Processing

2.1 Introduction

Group delay is a measure of time delay of the amplitude envelopes of the various sinusoidal components of a signal through a device under test, and is a function of the frequency. Recent research (Yeşil et al., 1984, 1991; Murthy and Yeşilnarayana, 1991b,a; Prasad et al., 2004; Rapisuram et al., 2008b; Dutta and Murthy, 2014; Gulati et al., 2014; Sebastian et al., 2017) has shown that group delay functions are effective in processing various types of signals ranging for speech, music and brain signals.

In this chapter, a review of group delay functions is presented. Section 2.2 discusses group delay and its properties. Modified group delay functions are discussed in Section 2.3, followed by modified group delay analysis on flattened power spectrum in Section 2.4. Section 2.5 discusses few applications of modified group delay functions in speech and music signal processing.

2.2 Group delay in speech signal processing

A complete representation of a signal in the Fourier domain requires both the Fourier transform magnitude and Fourier transform phase. Earlier studies have already established the significance of the short-term phase spectrum in speech processing applications. The resonances of the vocal tract manifest as peaks in the short-term magnitude spectrum of the speech signal. These resonances often called formants, manifest as transitions in the short-time phase spectrum. These transitions are masked due to the phase wrapping at multiple of $2\pi$ as illustrated in Figure 2.1 (a). Hence any meaningful use of the short-time phase spectrum for speech processing involves the process of phase unwrapping (Figure 2.1 (b)). Group delay is defined as the negative derivative of
the unwrapped Fourier transform phase. Group delay computed for a signal is shown in 2.1 (c). Mathematically the group delay function, \( \tau_p(e^{j\omega}) \) is given by,

\[
\tau_p(e^{j\omega}) = -\frac{d\{\arg(X(e^{j\omega}))\}}{d\omega}.
\] (2.1)

where \( \omega \) is the angular frequency, \( X(e^{j\omega}) \) is the Fourier transform of the signal \( x[n] \) and \( \arg(X(e^{j\omega})) \) is the phase function. Similarly, a group delay function \( \tau_m(e^{j\omega}) \) derived from Fourier transform magnitude function \( |X(\omega)| \) can also be found in (Yegnanarayana et al., 1984).

Consider a discrete time signal \( x[n] \). Then

\[
X(e^{j\omega}) = |X(e^{j\omega})|e^{j\arg(X(e^{j\omega}))}
\] (2.2)

Taking logarithm on both sides,

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

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

where, \( \text{Im} \) corresponds to the imaginary part. The group delay function can be computed by (Oppenheim and Schafer, 1990; Yegnanarayana et al., 1984),

\[
\tau(e^{j\omega}) = -\text{Im}\frac{d(\log(X(e^{j\omega})))}{d\omega}
\] (2.5)

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

where the subscripts \( R \) and \( I \) denote the real and imaginary parts, respectively. \( X(e^{j\omega}) \) and \( Y(e^{j\omega}) \) are the Fourier transforms of \( x[n] \) and \( nx[n] \), respectively. \( |X(e^{j\omega})|^2 \) in the denominator, makes the group delay function noisy (Murthy, 1991). Hence group delay functions are modified to eliminate the effects of this spikes to generate modified group delay functions (Murthy, 1991; Hegde, Rajesh M., 2005).

Using Equation 2.2 (and assuming that the signal is minimum phase), it can also be shown that (Oppenheim and Schafer, 1990; Yegnanarayana et al., 1984; Murthy and Yegnanarayana, 2011)
\[ \ln |X(e^{j\omega})| = c[0]/2 + \sum_{n=1}^{\infty} nc[n] \cos n\omega \] (2.7)

and the unwrapped phase function is given by

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

where \(c[n]\) are the cepstral coefficients. Taking the negative derivative of Equation-2.8 with respect to \(\omega\), we get

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

From Equations 2.8 and 2.9, it is worth noting that, for a minimum phase signal, the spectral phase and magnitude are related through the cepstral coefficients. Further, the group delay function can be obtained as the Fourier transform of the weighted cepstrum.
2.2.1 Properties of group delay functions

The group delay functions and their properties have been discussed extensively in (Murthy and Yegnanarayana, 2011). The basic properties are listed below.

- Poles (Zeros) of the transfer function show as peaks (valleys) in the group delay function.
- For a minimum phase signal
  \[ \tau_p(\omega) = \tau_m(\omega) \]  \hspace{1cm} (2.10)
- For maximum phase signal
  \[ \tau_p(\omega) = -\tau_m(\omega) \]  \hspace{1cm} (2.11)
- For a mixed phase signal
  \[ \tau_p(\omega) \neq \tau_m(\omega) \]  \hspace{1cm} (2.12)

Figure 2.2: Illustration of additive property of the group delay functions. Three signals \(x[n], h[n] \) and \( y[n] \), where \( y[n] = x[n] * h[n] \) (upper pane); magnitude spectra of three signals (middle pane); group delay functions of three signals (the lower pane).
Additive property

The group delay function exhibits additive property. Convolution of signals in the time domain is reflected as a summation in the group delay domain.

Consider a linear time invariant (LTI) system with impulse response given by

\[ H(e^{j\omega}) = H_1(e^{j\omega})H_2(e^{j\omega}) \]  \hspace{1cm} (2.13)

where \( H_1(e^{j\omega}) \) and \( H_1(e^{j\omega}) \) are the responses of the two resonators whose product gives the overall system response. Taking absolute value on both sides we have,

\[ |H(e^{j\omega})| = |H_1(e^{j\omega})||H_2(e^{j\omega})| \]  \hspace{1cm} (2.14)

Using the additive property of the Fourier Transform phase,

\[ \text{arg}(H(e^{j\omega})) = \text{arg}(H_1(e^{j\omega})) + \text{arg}(H_2(e^{j\omega})) \]  \hspace{1cm} (2.15)

Then the overall group delay function \( \tau_h(e^{j\omega}) \) is given by,

\[ \tau_h(e^{j\omega}) = -\frac{\partial(\text{arg}(H_1(e^{j\omega})))}{\partial \omega} - \frac{\partial(\text{arg}(H_2(e^{j\omega})))}{\partial \omega} \]  \hspace{1cm} (2.16)

\[ = \tau_{h_1}(e^{j\omega}) + \tau_{h_2}(e^{j\omega}) \]  \hspace{1cm} (2.17)

Additive property can be observed in the example shown in Figure 2.2. Consider two time domain signals \( x[n] \), \( h[n] \) and the convolution of \( x[n] \) and \( h[n] \) (denoted by \( y[n] \)), shown in the upper pane in Figure 2.2. The middle pane in Figure 2.2 shows corresponding magnitude spectra. The lower pane illustrates the additive property of group delay functions.

High resolution property

The group delay function has a higher resolving power as compared to the magnitude spectrum. The high-resolution property of group delay functions is illustrated in Figure 2.3 (Sebastian et al., 2015a). The faster decay of group delay in comparison to magnitude spectrum, shows the high resolving power of group delay functions.
Figure 2.3: Illustrating the high-resolution property for a two conjugate pole system. The faster decay of group delay in comparison to magnitude spectrum can be seen.

The high resolution property enables group delay function to estimate the frequency components present in a composite signal. Figure 2.4 (a) shows a composite signal comprising two frequency components. Figure 2.4 (b), (c) and (d) represent magnitude spectrum, group delay and modified group delay spectrum for a frame, respectively. Two prominent peaks are observed corresponding to the two components of the signal in group delay spectrum (Figure 2.4 (d)).

Figure 2.4: (a) Noisy composite signal, (b) Magnitude Spectrum, (c) Group delay spectrum of (a), (d) Modified group delay spectrum of (a).
A \( z \)-plane plot of a system consisting of two complex conjugate pole pairs is shown in Figure 2.5 (a). The corresponding magnitude and group delay spectrum are shown in Figure 2.5 (b) and 2.5 (c), respectively. Two peaks are well resolved in group delay domain. The ability of the group delay function to resolve closely spaced formants in the speech spectrum is explained in (Murthy and Yegnanarayana, 1991a). In (Kumar, 2015), examples of single-resonator and multi-resonator systems are considered and the resolution of group delay and magnitude spectrum are studied. Using kurtosis and spectral flatness as measures, it is shown that the group delay function exhibits greater peakedness compared to that of the magnitude spectrum.

Consider a causal, discrete time signal \( x[n] \), whose \( Z \)-transform \( X(z) \) is given by a 4\(^{th} \) order all pole model defined by:

\[
X(z) = \frac{1}{(z - z_0^*)(z - z_0)(z - z_1^*)(z - z_1)} \tag{2.18}
\]

where * indicates complex conjugation. \( z_i = e^{-\sigma_i + j\omega_i} \), where \( e^{-\sigma_0}, e^{-\sigma_1} \) determines proximity of the root to unit circle.

\[
X(e^{j\omega}) = \prod_{i=0}^{1} \frac{1}{(e^{j\omega} - e^{-(\sigma_i - j\omega_i)})} \prod_{i=0}^{1} \frac{1}{(e^{j\omega} - e^{-(\sigma_i + j\omega_i)})}. \tag{2.19}
\]

The phase spectrum of the system defined by Equation 2.18 is given by,

\[
\theta(\omega) = -\sum_{i=0}^{1} \tan^{-1} \left( \frac{\sin \omega - e^{-\sigma_i} \sin \omega_i}{\cos \omega - e^{-\sigma_i} \cos \omega_i} \right) + \\
-\sum_{i=0}^{1} \tan^{-1} \left( \frac{\sin \omega + e^{-\sigma_i} \sin \omega_i}{\cos \omega - e^{-\sigma_i} \cos \omega_i} \right). \tag{2.20}
\]

Differentiating Equation 2.20 with respect \( \omega \), it can be shown that:

\[
\theta'(\omega) = \theta'_1(\omega) + \theta'_2(\omega) + \theta'_3(\omega) + \theta'_4(\omega) \\
\tau(\omega) = \tau_1(\omega) + \tau_2(\omega) + \tau_3(\omega) + \tau_4(\omega) \tag{2.21}
\]

where \( \theta'_i \) corresponds to derivative of each of the terms in Equation 2.20, and \( \tau_i(\omega) \) corresponds to the group delay functions of each of the first order poles, respectively.

For simplicity, we consider group delay function of the first two terms alone. For sake
of argument, let $\sigma_1 = 3 \sigma_0$

$$
\tau_+(\omega) = -\frac{1 - e^{-\sigma_0} \cos(\omega - \omega_0)}{1 + e^{-2 \sigma_0} - 2 e^{-\sigma_0} \cos(\omega - \omega_0)} + \frac{1 - e^{-3 \sigma_0} \cos(\omega - \omega_1)}{1 + e^{-6 \sigma_0} - 2 e^{-3 \sigma_0} \cos(\omega - \omega_1)}.
$$

Taking the derivative of $\tau_+(\omega)$ and setting it to zero, we have

$$
\tau'_+(\omega) = \frac{(e^{-\sigma_0} - e^{-3 \sigma_0}) \sin(\omega - \omega_0)}{(1 + e^{-2 \sigma_0} - 2 e^{-\sigma_0} \cos(\omega - \omega_0))^2} + \frac{(e^{-3 \sigma_0} - e^{-9 \sigma_0}) \sin(\omega - \omega_1)}{(1 + e^{-6 \sigma_0} - 2 e^{-3 \sigma_0} \cos(\omega - \omega_1))^2} = 0
$$

When $\sigma_0 = 0$, Equation 2.23 becomes identically zero. This corresponds to zeros or poles that lie exactly on the unit circle, for example, the everlasting sinusoids. At $\omega = \omega_0$, first term becomes zero, while the second term takes values that are $<< 1$ owing to the factor $(e^{-3 \sigma_0} - e^{-9 \sigma_0})$. Owing to the $\cos$ and $(e^{-3 \sigma_0} - e^{-9 \sigma_0})$, the group delay function is symmetric and decays fast about the resonant frequency $\omega_0$. A similar observation can be made when $\omega = \omega_1$, as long as $\sigma_0 > 0$ and $\omega - \omega_0, \omega - \omega_1$ are less than $\pi$. The narrow width around the resonant frequency and the relation between the height and bandwidth of a resonance is referred to as the high resolution property.
2.3 Modified group delay function (MODGD)

It has been shown that group delay functions can be used to accurately represent signal information as long as the roots of the z-transform of the signal are not too close to the unit circle in the z-plane (Madhumurthy and Yegnanarayana, 1989). The group delay function exhibits large spikes due to zeros that are very near to the unit circle. For speech signals, the zeros are introduced by pitch, nasals, the glottal return phase and the short-time analysis. Hence, the group delay function has to be modified to eliminate the effects of these spikes. The group delay function is modified by replacing $|X(e^{j\omega})|$ in the denominator by its cepstral smoothed envelope $|S(e^{j\omega})|$. The significance of cepstral smoothing is explained in detail in (Hegde, Rajesh M., 2005). The Equation 2.6 is modified to obtain $\tau_c(e^{j\omega})$ as,

$$\tau_c(e^{j\omega}) = \frac{X_R(e^{j\omega})Y_R(e^{j\omega}) + Y_I(e^{j\omega})X_I(e^{j\omega})}{|S(e^{j\omega})|^2}$$ \hspace{1cm} (2.24)

Since the peaks at the formant locations are very spiky in nature, parameters are introduced to reduce the amplitude of these peaks and to restore the dynamic range of the speech spectrum. The algorithm for computation of the modified group delay function is described in (Hegde et al., 2007b) and it can be obtained as,

$$\tau_m(e^{j\omega}) = \left(\frac{\tau(e^{j\omega})}{|\tau(e^{j\omega})|}\right)^\alpha |\tau(e^{j\omega})|^\gamma$$ \hspace{1cm} (2.25)

where,

$$\tau(e^{j\omega}) = \frac{X_R(e^{j\omega})Y_R(e^{j\omega}) + Y_I(e^{j\omega})X_I(e^{j\omega})}{|S(e^{j\omega})|^2}.$$ \hspace{1cm} (2.26)

Two new parameters, $\alpha$ and $\gamma$ are introduced to control the dynamic range of MODGD such that $0 < \alpha \leq 1$ and $0 < \gamma \leq 1$ (Murthy, 1991; Hegde, Rajesh M., 2005). The algorithm is summarized in the Table 2.1. Modified group delay analysis performed directly on the speech signal is henceforth denoted by MODGD (Direct).

The pole-zero plot of a system characterized by two poles and their complex conjugates is shown in Figure 2.6 (a). The corresponding group delay spectrum is shown in Figure 2.6 (b). In Figure 2.6 (c), the pole-zero plot of the same system is shown with zeros added uniformly in very close proximity to the unit circle. It is evident from Figure 2.6 (d), that the group delay spectrum for such a system becomes very spiky and
ill defined. In Figure 2.6 (e), all the zeros are manually moved (radially) into the unit circle and the group delay function of such a system is recomputed. The group delay spectrum of the system in Figure 2.6 (e) is shown in Figure 2.6 (f).

Table 2.1: Algorithm for the computation of MODGDF

<table>
<thead>
<tr>
<th>Steps</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Let ( x[n] ) be the given sequence.</td>
</tr>
<tr>
<td>2</td>
<td>Compute the DFT ( X[k], Y[k], ) of ( x[n] ) and ( nx[n] ), respectively.</td>
</tr>
<tr>
<td>3</td>
<td>Group delay function is ( \tau_x[k] = \frac{X_R[k]Y_R[k]+X_I[k]Y_I[k]}{</td>
</tr>
<tr>
<td>4</td>
<td>Compute modified group delay ( \tau_c[k] = \frac{X_R[k]Y_R[k]+X_I[k]Y_I[k]}{</td>
</tr>
</tbody>
</table>
| 5     | Two new parameters \( \alpha \) and \( \gamma \) are introduced in the equation to convert to modified group delay function, \( \tau_m[k] \) by,

\[
\tau_m[k] = \frac{\tau[k]}{|\tau[k]|^\alpha}
\]

\[
\tau[k] = \frac{X_R[k]Y_R[k]+X_I[k]Y_I[k]}{|S[k]|^2}
\]

Figure 2.6: (a) The \( z \)-plane with two conjugate pole pairs, (b) The group delay spectrum of the system shown in (a), (c) The \( z \)-plane with two pole pairs and zeros added, (d) the group delay spectrum of the system shown in (c), (e) The \( z \)-plane with zeros pushed radially inward, (f) the group delay spectrum of the system shown in (e).
Modified group delay function as a feature (MODGDF)

The modified group delay function has a dimension that is equal to that of the order of the discrete Fourier transform (DFT) chosen. The dimension is reduced by transforming the MODGD spectrum via the discrete cosine transform (DCT) to the cepstral domain. The DCT is primarily used as a data independent decorrelator (Yip and Rao, 1997). DCT is used to convert the modified group delay spectra into cepstral features, \( c[n] \) given by,

\[
c[n] = \sum_{k=0}^{k=N_f} \tau_m[k] \cos\left(\frac{n(2k + 1)\pi}{N_f}\right)
\]

where \( N_f \) is the DFT order and \( \tau_m[k] \) is the modified group delay spectrum. This is known as modified group delay feature MODGDF. The velocity and acceleration parameters for the new group delay function are defined in the cepstral domain, in a manner similar to that of the velocity and acceleration parameters for mel-frequency cepstral coefficient (MFCC).

Modified group delay features are used extensively in speech recognition tasks (Hegde et al., 2007a), and robust voice activity detection (Hari Krishnan P et al., 2006). The complementarity of the group delay features with respect to other conventional acoustic features is discussed in (Rasipuram et al., 2008b). In (Rasipuram et al., 2008b), the joint features formed by concatenating spectral magnitude based MFCC and spectral phase based MODGDF are used for recognizing phonetic unit classes and it has shown improvement in performance compared to other combinations. In (Kumar et al., 2010), automatic speech recognition using feature switching based on KL divergence is proposed. A significant improvement in performance is observed over the joint feature stream consisting of MFCC and MODGD.

Numerator of group delay functions

The characteristics of the vocal tract system can also be computed using a variant of group delay functions, namely numerator of group delay (Yegnanarayana et al., 2011). The idea of numerator of group delay is introduced in (Anand Joseph et al., 2006).

Group delay function can be expressed by (Anand Joseph et al., 2006),

\[
\tau(e^{j\omega}) = \frac{X_I(e^{j\omega})X_R'(e^{j\omega}) - X_R(e^{j\omega})X_I'(e^{j\omega})}{X_R(e^{j\omega})^2 + X_I(e^{j\omega})^2}.
\]
where $X(e^{j\omega}) = X_R(e^{j\omega}) + jX_I(e^{j\omega})$ is the Fourier transform of the discrete-time signal $x[n]$, and $X'(e^{j\omega}) = X'_R(e^{j\omega}) + jX'_I(e^{j\omega})$ is its derivative. The group delay function of a signal around the resonant frequency is proportional to the square of the magnitude of the Fourier transform, That is,

$$\tau(e^{j\omega}) \propto |X(e^{j\omega})|^2 \quad (2.29)$$

If we ignore the denominator term, and consider the numerator term alone in the speech signal, numerator of group delay function can be expressed by,

$$g(e^{j\omega}) = X_I(e^{j\omega})X'_R(e^{j\omega}) - X_R(e^{j\omega})X'_I(e^{j\omega}) \quad (2.30)$$

From Equation 2.28, we get,

$$g(e^{j\omega}) = \tau(e^{j\omega})|X(e^{j\omega})|^2 \quad (2.31)$$

Using the Expression 2.29, we can conclude that

$$g(e^{j\omega}) \propto |X(e^{j\omega})|^4 \quad (2.32)$$

Thus, the numerator of group delay shows sharper peaks near the resonances than $\tau(e^{j\omega})$ and it has been successfully employed in formant estimation (Anand Joseph et al., 2006) and automatic speech recognition (Zhu and Paliwal, 2004).

2.4 Modified group delay on flattened spectrum

MODGD (Source)

In the previous section, we have seen the group delay analysis on the speech signal directly, MODGD (Direct). In this section, group delay on the flattened spectrum - MODGD (Source) is explained. The power spectrum of the speech signal is flattened using root cepstral smoothing (Murthy, 1994). A flat power spectrum is first obtained by dividing the spectrum by its envelope. What remains are the picket fence harmonics corresponding to that of the pitch of the source. This signal resembles a sinusoid in noise. The modified group delay function computed for this signal is defined as
Figure 2.7: (a) Frame of synthetic signal, (b) Magnitude spectrum, (c) Smoothened spectrum, (d) Flattened spectrum.

MODGD (Source). The group delay spectrum produces peaks at multiples of pitch period, $T_0$. Figure 2.7 illustrates the process of flattening the spectrum for a synthetic signal frame. It is observed that the harmonics due to pitch are clearly visible in the flattened spectrum (Figure 2.7 (d)).

### Cepstral smoothing

Cepstral smoothing approach for spectrum flattening is illustrated in Figure 2.8. In the cepstral domain, the signal is decomposed into the spectral envelope (lower order cepstral coefficients) and the spectral fine structure (higher order cepstral coefficients). Speech is mainly represented by the lower order cepstral coefficients and a cepstral peak in the upper order cepstral coefficients represents the pitch information.

![Cepstrum method of smoothing](image)

Figure 2.8: Cepstrum method of smoothing

The smooth spectral envelope can be obtained by cepstral smoothing using the steps
shown in Figure 2.8. The spectral envelope obtained by cepstral smoothing is,

$$Y_m = \text{DFT} \left[ w \cdot \text{DFT}^{-1} \log(|X_m|) \right]$$

where \( w \) and \( X_m \) represent low pass lifter in the cepstral domain and DFT of \( x[n] \), respectively. \( w \) can be expressed by,

$$w[n] = \begin{cases} 1, & |n| < n_c \\ 0.5, & |n| = n_c \\ 0, & |n| > n_c \end{cases}$$

where, \( n_c \) determines the liftering window length. The log magnitude spectrum of \( X(e^{j\omega}) \) is thus low pass filtered to obtain a smooth spectral envelope.

**Root cepstrum method**

The root cepstrum based method is a non-model based approach which is similar to the conventional cepstrum approach, except that the logarithmic operation is replaced by the \( | \cdot |^{\gamma} \) operation, with \( 0 < \gamma \ll 1 \) (Murthy, 1994). The root cepstrum method is shown in Figure 2.9. Root cepstral based smoothing is preferred as it not only helps in capturing the dynamic range of the spectrum but also de-emphasizes the convolution effects of the window function in the frequency domain.

**2.5 Group delay applications in speech and music**

The potential of group delay functions has been exploited in speech and music signal processing applications. Group delay function has already established its role in pitch/formant estimation, voice activity detection, segmentation of speech into syllable
boundaries, speech emotion detection and text-to-speech synthesis. The effectiveness of segmentation of speech, and the features derived from the modified group delay functions are demonstrated in applications such as language identification (Nagarajan and Murthy, 2006), speech recognition (Hegde et al., 2007b) and speaker identification (R.M. Hegde et al., 2004). MIR applications also employ group delay functions in tonic identification, motif recognition and discovery, onset detection and audio melody extraction tasks. Few applications are briefly discussed in the following section.

2.5.1 Pitch and formant estimation

Additive and high resolution properties of group delay functions are exploited in estimating system/source parameters. Group delay analysis on the flattened spectrum can be used for pitch estimation (Murthy, 1991). The performance of the proposed algorithm is superior to many algorithms especially, in noisy environment. Pitch and first formant frequency computed for a speech utterance, using modified group delay functions are shown in Figure 2.10. Wavesurfer (W.S-URL, 2012) references are also given. The work proposed in (Sebastian et al., 2015b) explores the use of peakedness and high resolution properties of the group delay functions and the ability of grating compression transform (GCT) to smear harmonically related components in the spectrum to track pitch across frame. Modified group delay functions are converted to useful cepstral features (MODGDF) and it is used effectively in speaker recognition and verification tasks (Hegde, Rajesh M., 2005). In (Rao et al., 2007), the instants of significant excitation in speech signals are computed using Hilbert envelope and group delay function. In the first phase, the approximate epoch locations are determined using the Hilbert envelope of the linear prediction residual. Later, the accurate locations of the instants of significant excitation are determined by computing the group delay around the approximate epoch locations. A group delay based method for epoch extraction from speech is proposed in (Murthy and Yegnanarayana, 2008). The average slope of the unwrapped phase of the short-time Fourier transform of LP residual is computed. Instants, where the phase-slope function makes a positive zero-crossing are identified as epochs. In (Murthy and Yegnanarayana, 1991.), a spectral root group delay function approach is proposed for formant estimation. The algorithm is similar to the cepstral smoothing approach for formant extraction using homomorphic deconvolution (Murthy and Yegnanarayana, 2011).
Figure 2.10: (a) A speech utterance, (b) First formant frequency estimated using MODGD, (c) Pitch estimated using MODGD (Wavesurfer reference is also given).

2.5.2 Speech and speaker recognition

Group delay functions are utilized in speech recognition task using two approaches (Kumar and Murthy, 2009; Rasipuram et al., 2008a; Padmanabhan and Murthy, 2010; Murthy and Yegnanarayana, 2011). In the first approach, segmentation algorithm is used to segment the speech signal at syllable-level. These boundaries are used in a hidden Markov model (HMM) framework for two different tasks, namely, speech recognition and synthesis. In speech recognition, the boundaries are used to reduce the search space for the language model, while in speech synthesis, the boundaries are used to obtain accurate phoneme boundaries by restricting embedded re-estimation in the HMM framework to syllable boundaries. In another approach, the modified group delay function is converted into features and subsequently used for speech recognition using machine learning algorithms. In (Sarada et al., 2009), a method is proposed to automatically segment and label continuous speech signal into syllable-like units for Indian languages. In this approach, the continuous speech signal is first automatically segmented into syllable-like units using group delay based algorithm. Similar syllable segments are then grouped together using an unsupervised and incremental training (UIT) technique.
The complementary nature of the MODGDF derived from the modified group delay function with respect to features derived from the Fourier transform magnitude spectra is illustrated with the help of extensive experiments in (Hegde, Rajesh M., 2005). The work proposed in (Dey et al., 2011) explores the use of feature-switching framework in speaker recognition task. The work shows the ability of MODGDF in capturing information complementary to the mel-frequency cepstral coefficients and linear predictive cepstral coefficients (LPCC). Sub-band approach proposed in (Thiruvaran et al., 2007) to restrict the masking effect, shows a competitive performance in recognition task as compared to other approaches. The group delay functions derived from parametric all-pole models are also used in speaker identification (Rajan et al., 2013). In (Bastys et al., 2010), phase spectrum of all-pole linear prediction model is used to derive features for speaker identification. The results are reported using RUSsian speech dataBASE (RUSBASE) with competitive performance as compared to the features derived from the power spectrum.

2.5.3 Speech synthesis

Concatenative speech synthesis using unit selection approach relies on a large database of basic units. The idea behind unit selection synthesis is to select the best sequence of speech units from all possible contexts from a database of speech units. Group delay based segmentation has already successfully been employed for segmenting the speech signal at syllable-level (Nagarajan et al., 2003). Segmenting the speech at syllable-level is well suited for Indian languages, which are syllable-timed. The algorithm is modified using vowel onset point detection to reduce insertion and deletion errors. A labeling tool is developed using this idea and was successfully implemented in six Indian Languages, namely, Tamil, Hindi, Bengali, Malayalam Telugu and Marathi (Deivapalan et al., 2008). Group delay based segmentation of syllables has been employed for building syllable based text-to-speech systems in (Pradhan et al., 2013) and (Vinodh et al., 2010). An automatic algorithm for phoneme segmentation on HTS is proposed in (Shanmugam and Murthy, 2014). In (Shanmugam and Murthy, 2014), group delay based processing of short-time energy (STE) is used in tandem with HMM based forced Viterbi algorithm to get accurate syllable boundaries. The group delay based boundaries obtained in the vicinity of the HMM syllable boundaries are used as correct boundaries to reestimate the monophone HMM models, where the monophone HMMs are
restricted to the syllable boundaries rather than the whole utterance. The reestimated boundaries are again compared with the group delay boundaries and are corrected again (Shanmugam and Murthy, 2014).

### 2.5.4 Speech emotion recognition

Nowadays, increasing attention has been directed to identify the emotional content of a spoken utterance. It is being employed in numerous applications such as humanoid robots, car industry, call centers, mobile communication and computer tutorial applications. Proper mapping of acoustic cues in terms of features has an important role in speech emotion recognition systems. A novel feature based on the phase response of an all-pole model of the vocal tract obtained from linear predictive coefficients (LPC) is used in (Sethu et al., 2007). The performance of group delay features for depression detection was also addressed (Ming et al., 2013). The experimental results demonstrate the potential of mel-frequency delta-phase (MFDP) over MFCC. The feature switching paradigm has been also implemented in speech emotion recognition systems (Dey et al., 2011). The underlying principle behind the method is that some features are better at discriminating some classes and other features for other classes. The early fusion of MFCC and MODGDF has shown superior performance than individual feature set in recognizing emotions, anger, boredom and anxiety. An information theoretic procedure is described in the paper which can be used to determine the feature most suitable for a given class.

### 2.5.5 Voice activity detection

Voice activity detection (VAD) also known as speech activity detection detects the presence or absence of human speech in an audio signal. VAD is an important enabling technology for a variety of speech-based applications such as speech coding, speech recognition, echo cancellation, speech enhancement and simultaneous voice/data applications. Voice activity detection using group delay is proposed in (Hari Krishnan P et al., 2006). The speech regions of the signal are characterized by well-defined peaks in the group delay spectrum, while the non-speech regions are identified by well-defined valleys. Two modified group delay function based VAD algorithms, one using Gaussian mixture models and the other using multilayer perceptrons are discussed in (Pad-
manabahan, 2012). The evaluation of the proposed algorithms shows a superior performance over standard VAD algorithms, with average error reduction of about 8%. The proposed work in (Kemp, T. and Waibel, A., 1999), investigates the joint use of source and filter-based features in voice activity detection. A mutual information-based assessment shows superior discrimination power of chirp group delay (CGD) of the zero-phase signal in voice activity detection.

2.5.6 Chirp group delay processing and ASR

The chirp group delay functions, in which group delay is computed on a circle other than the unit circle, are proposed in (Bozkurt et al., 2007). The negative derivative of the phase spectrum computed from chirp z-transform is termed as chirp group delay (Bozkurt et al., 2007). The usefulness of chirp group delay functions in feature extraction is demonstrated for automatic speech recognition (ASR) in (Bozkurt et al., 2007). In (Jayesh and Ramalingam, 2014), a variant of chirp group delay analysis is proposed, which is useful in better vocal tract estimation. Zeros outside the unit circle are first reflected inside the unit circle. The chirp group delay function of the transformed signal is computed. It does not require knowledge of glottis closure instants (GCI) and is much less sensitive to starting point of the analysis window and its duration.

2.5.7 Tonic identification

Group delay functions have also found many applications in music processing. Automatic tonic identification in Carnatic music has a major role in performing data driven computational melodic analysis (Bellur and Murthy, 2013). The tonic is the base pitch chosen by a performer, which serves as a reference throughout a performance. Group delay function is employed in the work proposed by Ashwin et al. (Bellur, 2013) to process pitch histograms. Three methods, namely concert based method, template matching and segmented histogram are proposed in (Bellur, 2013). In concert based method, group delay histograms computed over all individual items are multiplied bin-wise to one single histogram. The peak with maximum value in the histogram is mapped to tonic pitch. The template matching method attempts to fit a template for every peak of the histogram, each of which is treated as a candidate Sa. The best template match helps to estimate the tonic pitch. In the third method, picking the global peak in the bin-wise
product of the segmented group delay histogram results in tonic pitch. Amongst the template matching and segmented histograms methods, template matching method is found to perform better.

### 2.5.8 Onset detection

Bello et al. define an onset as the instant chosen to mark the transient (Bello et al., 2005). Specifically in (Kumar et al., 2015), a method is proposed to identify stroke instants in the case of percussive instruments. A novel algorithm for onset detection which utilizes high spectral resolution of group delay functions is presented in (Kumar et al., 2015). It employs the resemblance of Carnatic percussion instruments to a generic amplitude-frequency modulated waveform. Group delay response of this signal is treated as a detection function in this work. The algorithm is observed to perform better on the mridangam which contains a considerable number of silent strokes. Two improvisations are proposed; one removes the parameter dependency, and the other extends group delay processing to a more generic onset detection algorithm. Andre Holzapfel et al. introduce a novel approach to estimate onsets in musical signals using the average of the group delay function (Holzapfel and Stylianou, 2008a). In this work, onsets are detected by locating the positive zero-crossings of the phase slope function.

### Summary

Group delay functions and the need for modified group delay functions are discussed in this chapter. Two important properties of modified group delay functions; additive property and high resolution property are explained with illustrations. Two variants of modified group delay functions, MODGD (Direct) and MODGD (Source) are explained. Finally, a review on various applications of modified group delay function in speech and music signal processing is given. These applications show the potential of modified group delay functions in speech-music analysis and content based information retrieval.
CHAPTER 3

Melody Extraction in Polyphonic Music

3.1 Introduction

In many cultures, music is an important part of people’s way of life, as it plays a key role in religious rituals, rite of passage ceremonies, social activities and cultural activities ranging from amateur karaoke singing to community choir.\(^1\) Music is an important performance art form conveyed through sound and silence\(^2\). The common elements of music include melodic pitch and rhythm. A variety of styles/genres of music have evolved owing to the differences in these elements. Melody extraction is important for music information retrieval tasks. In this chapter, the task of audio melody extraction, evaluation framework and challenges are discussed. Two schemes for melody extraction, both based on modified group delay functions are proposed and evaluated on various datasets. Section 3.2 introduces the concept of music information retrieval system. The task of audio melody extraction is discussed in Section 3.3, followed by different approaches in melody extraction in Section 3.4. The methodology used in the performance evaluation of the algorithms is briefly discussed in Section 3.5. The challenges in melody extraction from polyphonic music are described in Section 3.6. The motivation for the use of modified group delay functions in audio melody extraction from music is listed in Section 3.7. The first scheme for melody extraction based on modified group delay functions is discussed in Section 3.8 followed by a revised algorithm in Section 3.9. Finally, the complementary nature of melodic feature is experimentally shown using standard dataset for music genre classification in Section 3.10.

3.2 Music information retrieval (MIR)

MIR mainly focuses on the understanding and usefulness of music data through research, development and application of computational approaches or tools. MIR in-

\(^{1}\)https://en.wikipedia.org/wiki/Music
\(^{2}\)https://cobussen.com/publications-and-courses/
cludes numerous tasks such as query-by humming (QBH), automatic genre classification, singing voice/instrument melody separation, audio melody extraction, instrument recognition and automatic music transcription. There are two main approaches to MIR;

Figure 3.1: Music information retrieval system

metadata-based and content-based (Wiering, 2006). In the former, the issue is mainly to find useful categories for describing music, distinguishing different recordings of the same composition or to organize the many genres that exist across the globe. Music information retrieval, when the music is associated with text tags, is a little easier than music that is not tagged. Retrieval using only the audio content is still a topical problem in MIR. Content-based MIR enables to find music that is similar to a set of features or an example (Wiering, 2006). The main components of a content based MIR system are shown in Figure 3.1. These components are query formation, description extraction, matching and music document retrieval. Musical information systems can be classified as per the scale of specificity (Casey et al., 2008). Systems with high specificity demand exact matching of the query and the content retrieved. In systems with low specificity, a query track will return tracks having little content directly in common with the query, but with some global characteristics that match. Mid-specificity systems match high-level music features, but do not match audio content. Table 3.1 enumerates some of the MIR use cases and their specificities. The proposed work for audio melody extraction which computes predominant melody from polyphonic music comes under the category of mid-specific systems.
Table 3.1: Music information retrieval and level of specificity

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Specificity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Music identification</td>
<td>H</td>
<td>Identify a compact disk, provide metadata about an unknown track, mobile music information retrieval</td>
</tr>
<tr>
<td>Plagiarism detection</td>
<td>H</td>
<td>Identify mis-attribution of musical performances, mis-appropriation of music intellectual property</td>
</tr>
<tr>
<td>Copyright monitoring</td>
<td>H</td>
<td>Monitor music broadcast for Copyright infringement or royalty collection</td>
</tr>
<tr>
<td>Versions</td>
<td>H/M</td>
<td>Remixes, live vs. studio recordings, Cover songs. Used for database normalization and near-duplicate results elimination</td>
</tr>
<tr>
<td>Melody</td>
<td>H/M</td>
<td>Find works containing a melodic fragment</td>
</tr>
<tr>
<td>Identical work/title</td>
<td>M</td>
<td>Retrieve performances of same OPUS or song title</td>
</tr>
<tr>
<td>Performer</td>
<td>M</td>
<td>Find music by a specific artist</td>
</tr>
<tr>
<td>Sounds like</td>
<td>M</td>
<td>Find music that sounds like a given recording</td>
</tr>
<tr>
<td>Performance alignment</td>
<td>M</td>
<td>Mapping one performance onto another independent of tempo and repetition structure</td>
</tr>
<tr>
<td>Composer</td>
<td>M</td>
<td>Find works by one composer</td>
</tr>
<tr>
<td>Recommendation</td>
<td>M/L</td>
<td>Find music that matches the user’s personal profile</td>
</tr>
<tr>
<td>Mood</td>
<td>L</td>
<td>Find music using emotional concepts: Joy, Energetic, Melancholy, Relaxing</td>
</tr>
<tr>
<td>Style/Genre</td>
<td>L</td>
<td>Find music that belongs to a generic category: Jazz, Funk, Female vocal</td>
</tr>
<tr>
<td>Instrument (s)</td>
<td>L</td>
<td>Find works with same instrumentation</td>
</tr>
<tr>
<td>Music-Speech</td>
<td>L</td>
<td>Radio broadcast segmentation</td>
</tr>
</tbody>
</table>

3.3 Melodic pitch estimation

Conventional pitch extraction algorithms used in the context of speech fail in music due to the presence of interfering partials. In music, unlike in speech, the pitch can span a number of octaves. The extraction of prominent pitch from polyphonic music signals has received substantial attention due to its relevance in music analysis. Prominent pitch or melodic pitch is the pitch contour of the lead or dominant musical instrument in the song. In a music with orchestration, there are several musical instruments playing along with singing voice which result in strong harmonic interferences. Predominant melody extraction becomes difficult owing not only to fast changes in pitch but also due to the presence of multiple voices. The multiple voices may be heterophonic \(^1\) as in Carnatic music or polyphonic \(^2\) as in Western music.

As per the widely accepted standard, melodic pitch is defined as “the single (monophonic) pitch sequence that a listener might reproduce when asked to hum or whistle a polyphonic piece of music” (Salamon and Gomez, 2012; Poliner et al., 2007). Histori-

\(^1\)en.wikipedia.org/wiki/Heterophony
\(^2\)en.wikipedia.org/wiki/Polyphony
cally, the term was derived from *melos*, which meant "song". In most songs, the melody follows a logical, mathematical pattern that creates a memorable line of notes. Melody extracted for a piece of music is shown in Figure 3.2. The importance of melody in music is explained well in (Field, 1998). It says “It is melody that enables us to distinguish one work from another. It is melody that human beings are innately able to reproduce by singing, humming, and whistling. It is melody that makes music memorable: we are likely to recall a tune long after we have forgotten its text”.

In polyphonic music, there are usually many instruments playing at the same time or with a lag. For example, different melodic sequences due to the accompaniments can be seen in Figure 3.3 with prominent melody emphasized in grey colour. Generally MIR community points to three definitions for melody (Salamon *et al.*, 2014a),

- The $f_0$ curve of the predominant melodic line drawn from a single source.
- The $f_0$ curve of the predominant melodic line drawn from multiple sources.
- The $f_0$ curves of all melodic lines drawn from multiple sources.

Definition 1 requires the choice of a lead instrument and gives the $f_0$ curve for this instrument. This annotation is the accepted standard in MIREX (Music Information
Retrieval Evaluation eXchange) evaluations. Definition 2 expands on definition 1 by allowing multiple instruments to contribute to melody. While a single lead instrument need not be chosen, an indication of which instrument is predominant at each point in time is required to resolve the $f_0$ curve to a single point at each time frame. Definition 3 is the most complex, but requires special attention in retrieval process. In a solo, the singing voice is the leading melody line. As per definition 1, melody extraction algorithm computes the pitch of singer as the prominent melody source. In fugue, two melody lines are present. A fugue begins with the exposition of its theme in one of the voices in the tonic key. After the statement of the theme, a second voice enters and states the theme transposed to another key, which is known as the answer. As per definition 2, the two voices contribute to the melody line for the entire performance. In Carnatic music, harmonium is used as a secondary melody instrument for vocal music. As per definition 3, two melody lines, one for the singer and the other for harmonium is computed simultaneously throughout the performance.

Melodic information can be conveyed through various representations (Zatorre and Baum, 2012). For example, consider a piece of music “Happy Birthday to You.”[^4]. Five different ways of representing this piece of music are illustrated in Figures 3.4 and 3.5. Melodic notation and metadata are shown in Figure 3.4. The metadata tag contains[^3][^4]

[^3]: https://en.wikipedia.org/wiki/Fugue
Figure 3.5: (a) Time domain signal, (b) melodic contour (c) spectrogram

information such as artist, album title, track number, genre, lyrics and more. Figure 3.5 (a), (b), and (c) show the time domain representation, the pitch contour and spectrogram representation, respectively for the same piece of music.

### 3.4 Melody extraction approaches

In the last two-three decades, audio melody extraction task has emerged as an active research topic in music retrieval, and numerous algorithms for the same have been proposed. Audio melody extraction is also known by several names such as predominant melody extraction, lead melody line computation and predominant fundamental frequency estimation (Salamon et al., 2014a). Existing melody extraction algorithms can be broadly classified into salience based approaches, source separation based approaches, salience and source separation based approaches, data-driven approaches and acoustic/musicological model based approaches.
3.4.1 Salience based approaches

A very successful approach to the task of melody extraction is salience based (Goto and Hayamizu, 1999; Cao et al., 2007; Hsu and Jang, 2010a). Salience based methods estimate potential $f_0$ candidates per frame based on pitch salience and apply tracking or transition rules to select the final melody line. In (Goto and Hayamizu, 1999), Goto et al. focus on the estimation of predominant musical voice rather than the transcription of all sound sources. The main steps of the melody extraction algorithm include selection of a candidate set of fundamental frequencies based on spectral peaks, and use of the subharmonic summation method to identify the fundamental frequency from this candidate set. This method estimates the relative dominance of every possible $f_0$ by maximum a posteriori probability (MAP) estimation. Cao et al. (Cao et al., 2007) propose a melody extraction method based on the subharmonic summation spectrum and the harmonic structure tracking strategy. They analyze the prominent pitch of the mixture to find stable harmonic structure seeds which are used to estimate pitch. By using the characteristics of vibrato and tremolo, Hsu and Jang (Hsu and Jang, 2010a) distinguish the vocal partial from the music accompaniment partials.

In another approach 5, the music signal is first preprocessed and autocorrelation function is used to form melodic fragments from the music. In the next step, these fragments are evaluated against each other and finally the melodic path is constructed such that it minimizes the steep changes in the tonal sequence. Salamon et al. propose a pitch extraction algorithm that extracts the pitch even when the accompaniment is strong (Salamon and Gomez, 2012). Pitch candidates are first estimated, and the context is used to determine pitch contours. Features such as length, height, pitch deviation and presence of vibrato are then used for framing rules for filtering non-melody segments.

3.4.2 Source separation based approaches

Another approach to melody extraction is, to use source separation methods (G.Hu and D.L.Wang, 2010; Wang and Brown, 2006; Richard et al., 2008; Durrieu et al., 2010; Tachibana et al., 2010). In source separation methods, the melody line is identified by separating it from the accompaniments. Source separation methods include (Richard et al., 2008), where an effort is made to separate the melody from polyphonic music.

---

5http://www.nyu.edu/classes/bello/ACA-files/4-periodicity.pdf
In this approach, source/filter model of the singing voice and a background model derived using non-negative matrix factorization (NMF) technique are used to estimate the fundamental frequency of the singing melody. (Durrieu et al., 2010) propose a new approach for the estimation and extraction of the main melody from polyphonic audio, where the leading vocal part is explicitly represented by a specific source/filter model. Two different representations are proposed for the source/filter model: a smooth instantaneous mixture model (SIMM) and a smooth Gaussian scaled mixture model (SGSMM). While the former represents the lead instrument as the instantaneous mixture of all possible notes, the latter allows one source/filter couple to be active at any moment. The final melodic pitch sequence is obtained by the Viterbi algorithm. The sustained and percussive sounds are separated using harmonic and percussive sound separation (HPSS) algorithm (Ono et al., 2008) in two passes in (Tachibana et al., 2010). From the retained enhanced components, the melodic pitch sequence is obtained directly from the spectrogram by finding the path which maximizes MAP of the frequency sequence.

### 3.4.3 Salience and Source separation based approaches

In this framework, source separation is first used as a preprocessing step to attenuate the accompaniment and then a salient function is computed from the processed signal to compute the melodic pitch sequence. Some of the methods which follow these steps can be found in (Hsu and Jang, 2010b; Tzu et al., 2012; Hsu et al., 2011). In (Hsu and Jang, 2010b), sinusoid partials are first extracted from the musical audio signal. The vibrato and tremolo information are then estimated for each partial. The vocal and instrument partials are discriminated according to a given threshold, and the instrument partials are then deleted. Instead of simply extracting the singing pitches by tracking the remaining vocal partials, the normalized harmonic summation spectrum (NHSS) is used as a salient function to track the potential candidates. Dynamic programming is later applied to find the melodic pitch contour. Yeh et al. propose a salience based method after accompaniments pruning (Tzu et al., 2012). In this method, a harmonic/percussive sound separation (HPSS) method is applied to suppress the energy produced by harmonic instruments. Later, sinusoidal partial extraction is performed by the multi-resolution fast Fourier transform (MR-FFT). A grouping method is adopted to combine partials of the main vocal and accompanying instruments. Then, vibrato and tremolo characteristics are used to prune the instrument partials. Finally, pitch trend is determined from the of
3.4.4 Data driven approaches

In (Poliner and Ellis, 2005; Ellis and Poliner, 2006; Bittner et al., 2005), a data driven approach is used wherein the entire short-time magnitude spectrum is used as training data for a support vector machine (SVM) classifier. The classifier is trained on thousands of spectral slices for which the suitable melody is found out through manual or human-corrected transcription of the original audio. In (Ellis and Poliner, 2006), a machine learning approach is proposed in which the system infers the correct melody label based only on training with labeled examples. In the proposed algorithm, each note is identified via SVM classifier trained directly from audio feature data, and the overall melody sequence is smoothed via a hidden Markov model (HMM). In (Bittner et al., 2005), the system uses pitch contours and contour features generated by MELODIA (Salamon and Gomez, 2012) for the task. This method consists of a contour labeling stage, a training stage where a discriminative classifier is used that distinguishes between melodic and nonmelodic contours, and a decoding stage which generates a final $f_0$ sequence. A classifier is used to score the pitch contours and remove those below a certain threshold. Viterbi decoding is adopted to get the final pitch trajectories. In (Ishwar, 2014), a combination of the present state-of-the-art systems and timbral characteristics of the various melodic sources is used to compute predominant pitch from audio music recording.

3.4.5 Acoustic/Musicological model based approaches

Some of the other prominent methods for melody extraction are described in (Ryynanen and Klapuri, 2008a; Rao and Shandilya, 2004). A method based on multiple $f_0$ estimation followed by acoustic and musicological modeling is proposed in (Ryynanen and Klapuri, 2008a). The acoustic model consists of separate models for singing notes and non-melody segments. The musicological model uses key estimation and note bigrams to determine the transition probabilities between notes. In (Rao and Shandilya, 2004), a pitch detection method based on a perceptual model is proposed to track voice pitch in the presence of a strong percussive background.
3.5 Evaluation methodology

Evaluation methodology is the same as that of MIREX evaluation standards. The performance of the melody pitch extraction for voiced frames is evaluated based on the following rule. The reference frequency of an unvoiced frame is considered to be 0 Hz. The estimated pitch of a voiced frame is considered correct, when it satisfies the following condition:

\[
| F_r(l) - F_e(l) | \leq \frac{1}{4} \text{tone (50 cents)}
\]

where \( F_r(l) \) and \( F_e(l) \) denote the reference frequency and the estimated pitch frequency on the \( l^{th} \) frame, respectively. 

Datasets

A music genre or sub-genre may be defined by the styles, the context, content and spirit of the themes. The method of melody extraction for one genre may not be optimal for a different genre. In MIREX evaluation, the performance of the algorithms is evaluated on a diverse collection of music styles. The music corpus includes ADC-2004, MIREX-05, MIREX-2008, LabROSA training set and MIR-1K dataset (Salamon and Gomez, 2012; Ramakrishnan et al., 2008).

Evaluation metrics

The performance of algorithms is conventionally evaluated using six metrics. Metrics defined in (Poliner et al., 2007) are as follows:

- **Voicing Recall Rate (VR)**: Proportion of frames labeled voiced in the ground truth that are estimated as voiced by the algorithm.

- **Voicing False Alarm Rate (VF)**: Proportion of frames labeled unvoiced in the ground truth that are estimated as voiced by the algorithm.

- **Raw Pitch Accuracy (RPA)**: The ratio between the number of the correct pitch frames in voiced segments and the number of all voiced frames.

---

6In the quarter tone scale, an octave is divided into 24 equal steps (equal temperament). In this scale, the quarter tone is the smallest step.
- **Raw Chroma Accuracy (RCA):** Same as raw pitch accuracy, except that both the estimated and ground truth $f_0$ sequences are mapped onto a single octave. RCA is greater than or equal to RPA.

- **Overall Accuracy (OA):** This measure combines the performance of the pitch estimation and voicing detection tasks to give an overall performance score for the algorithm. It is defined as the proportion of frames (out of the entire piece) correctly estimated by the algorithm, where for unvoiced frames, this means that the algorithm labeled them as unvoiced, and for voiced frames, the algorithm not only determined them correctly as voiced frames but also estimated $f_0$ accurately.

- **The Standard deviation of the pitch detection ($\sigma_e$):** It is defined as,

\[
\sigma_e = \sqrt{\frac{1}{N} \sum (p_s - p'_s)^2 - e^2} \tag{3.2}
\]

where $p_s$ is the standard pitch, $p'_s$ is the detected pitch, $N$ is the number of correct pitch frames and $e$ is the mean of the fine pitch error. $e$ is defined as:

\[
e = \frac{1}{N} \sum_N (p_s - p'_s). \tag{3.3}
\]

The accuracy and standard deviation measure the quality of the pitch estimation algorithms, while voicing recall and voicing false alarm refer to accuracy of the distinction between melody and non-melody segments. Overall accuracy is the overall average performance of the melody extraction algorithm.

### 3.6 Melody extraction: Challenges and Applications

There are a number of algorithms available in the literature for the melody extraction task. Nevertheless, many challenges still remain in the design and evaluation strategy (Salamon *et al.*, 2014a). Most of the algorithms developed so far mainly rely on the uniqueness of the human voice in audio melody extraction task. In an orchestration, where leading melody line is an instrument with other accompaniments, main melody line may be closer to the melody of accompanying instruments in many parts. In such situations, the task of distinguishing melody from the accompaniment is a challenging task. Another issue is the degree of polyphony in the evaluation material. Degree of polyphony is defined as the amount of orchestration as music styles vary. The errors that occur in melodic pitch estimation as the degree of polyphony increases are illustrated.
in Figure 3.6. In the figure, the melodic pitch estimated using pYIN algorithm (Mauch and Dixon, 2014) along with references are shown for the same song with varying orchestration. As degree of polyphony increases, melody estimation error also increases.

Octave error minimization techniques adopted by algorithms also play an important role in reducing spurious pitch estimates. Figure 3.7 shows the pitch contour of a music segment. The reference pitch contour is shown in red. It is obvious that in the circled part, the algorithm estimates pitch value as the double of the reference pitch values. In (Ramakrishnan et al., 2008), raw chroma accuracy of the algorithm is high due to occurrence of octave errors, which is attributed to biasing of the two way mismatch (TWM) error function towards pitches that lie in the lower and middle range regions of the \( f_0 \) search space. In the work proposed by Jo et al. (Jo and Yoo, 2010), the state transition is controlled by a factor called degree of melody line which makes the algorithm robust to octave mismatch. In (Paiva et al., 2006), octave correction stage is added to tackle errors that appear in the note elimination stage. The rule based post processing implementation helps to minimize the difference between RPA and RCA in the work proposed by Dressler (Dressler, 2011).

![Figure 3.6: Melodic contours with different degrees of polyphony. (a) low, (b) medium, and (c) high. (Melody is extracted using pYIN algorithm)](image)

Evaluation materials are required for assessing the performance of melody extraction algorithms. The duration of the audio clips, time offsets in the ground truth anno-
tations and the size/musical content of the databases also pose issues in evaluating the algorithmic performance of many systems. Audio clips currently used are too short to predict performance on full songs. Similarly, time offset between the annotation and the audio, can have a dramatic effect on the results (Salamon and Urbano, 2012).

![Figure 3.7: Octave errors in melodic pitch estimation](image)

Voicing detection stage always plays an important role in the overall performance. Till date, most approaches focus primarily on raw pitch extraction segment of melody extraction, and less so on the voicing decision. Currently, even the algorithms with the most effective voicing detection methods obtain an average voicing false alarm rate of more than 20%. The best result in melody-nonmelody segmentation in (Ellis and Poliner, 2006), are mainly due to the machine learning approach in the algorithm, which classifies dubious frames as non-melodic frames. The computational complexity of the algorithms is also very crucial in real time applications. Analysis shows that, the variation in computational time of various algorithms may be a hindrance in real time user-interactive applications (E.Gomez et al., 2006).

Melody extraction has a wide variety of application ranging from industry to research. It has already been shown that automatically extracted melodic pitch can be used in music retrieval (Salamon et al., 2013; Song et al., 2002; Ryynanen and Klapuri, 2008b; Qin et al., 2011; Tralie and Bendich, 2015), music classification (Su, 2012; Simsekli, 2010; Shan et al., 2002; Machado Rocha, 2011), music de-soloing (Jean-Marc and Christophe Ris, 1999; Ryynanen et al., 2008; Li and Wang, 2007), music transcription (Benetos et al., 2012; Bello et al., 2000; Gomez et al., 2012) and computational musicology (Pikrakis et al., 2012; Ishwar et al., 2013; Koduri et al., 2012). For example, melodic motif spotting using melodic contour is illustrated in Figure 3.8. The melodic
contour extracted for an ālāpana\textsuperscript{7} is shown in Figure 3.8. In the figure, two instances of the same melodic motif of rāga Kāmboji are circled in the melodic contour of the song.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.8.png}
\caption{Two instances of melodic motif of rāga Kāmboji in an ālāpana}
\end{figure}

3.7 Audio melody extraction from music using modified group delay functions

The primary motivation for this work arises from the effectiveness of group delay processing in speech and music (Yegnanarayana and Murthy, 1992; Rajan and Murthy, 2013; Vijayan et al., 2014). The audio melody extraction task consists of two tasks: (1) pitch detection (to extract the main melody in the presence of strong accompaniment) (2) voicing detection (to identify the presence or absence of the main melody). Most algorithms in the MIR literature are based on processing the harmonic structure of pitch in the Fourier transform magnitude spectrum for music (Salamon et al., 2014b). We propose two different approaches, where group delay processing is employed.

In the first approach, the modified group delay function (used earlier for formant extraction (Yegnanarayana \textit{et al.}, 1991)) is used for pitch extraction. The peak corresponding to the predominant pitch is emphasized in the modified group delay spectrum and is mapped to melodic pitch. This is explained in Section 3.8.

In the second approach, the starting point is the power spectrum of the signal. The power spectrum is subjected to group delay processing exploiting the additive and high

\textsuperscript{7}An ālāpana is a melodic improvisation within the permitted phraseology of a given melody.
resolution properties of group delay functions. The power spectrum is first flattened. The flattened spectrum appears to be periodic owing to the zeros due to pitch. This signal resembles a sinusoid in noise. Modified group delay function based processing of this signal is performed to estimate the frequency of the sinusoid which corresponds to pitch. A multi-resolution framework is incorporated to capture dynamic variation in melody. Dynamic programming is employed to ensure pitch consistency across frames. The removal of spurious pitch estimates in the post-processing step refines the pitch sequences, thus resulting in a smooth pitch trajectory. This is explained in Section 3.9.

3.8 Melodic-pitch estimation using MODGD (Direct)

The first approach in melodic pitch computation employs MODGD analysis on the music signal directly. In this approach, audio signals are frame-wise analyzed using modified group delay function. The proper selection of cepstral window emphasizes the peak corresponding to pitch in the group delay spectrum. The peaks in the range \([P_{\text{min}}, P_{\text{max}}]\) are subjected to consistency check using dynamic programming, where \(P_{\text{min}}\) and \(P_{\text{max}}\) represent minimum and maximum pitch, respectively. The optimal path obtained in the consistency check is mapped to predominant melodic pitch sequence. Finally, the pitch sequence is smoothed using median filtering to get a refined pitch contour in the ‘post-processing’ phase.

3.8.1 Theory of predominant melodic pitch estimation

A source-filter model for music is given by

\[
X(z) = H(z)E(z)
\]  

(3.4)

where \(E(z)\) corresponds to z-transform of the source and \(H(z)\) corresponds to that of the system spectrum. Using additive property of group delay, the overall group delay function, \(\tau_x(e^{j\omega})\) becomes

\[
\tau_x(e^{j\omega}) = \tau_h(e^{j\omega}) + \tau_e(e^{j\omega}).
\]  

(3.5)
where $\tau_h(e^{j\omega})$, $\tau_e(e^{j\omega})$ represent system group delay and source group delay, respectively.

![Figure 3.9: (a) Cepstral window for cepstral smoothing, (b) Peaks correspond to pitch and formants in MODGD plot.](image)

In the MODGDF computation, smoothed spectral envelope is obtained by cepstral smoothing. When the cepstral window is selected properly, a peak that corresponds to the pitch can be seen in the modified group delay function. The choice of shape and size of cepstral window is important. The purpose of the cepstral window is to ensure that the poles corresponding to the source are emphasized in the modified group delay spectrum. The source in speech or music is due to the glottal pulse. The glottal pulse produces a double pole at the pitch frequency. For the customization of the cepstral window, new parameters are introduced in the computation, namely $\delta$ (smoothing) and $\sigma$ (roll off), both varying from 0 to 1. For $\delta = 0$, window size is small and for $\delta = 1$, window size is large (no smoothing). Hanning window coefficients with parameters ($\delta$, $\sigma$) for the window size $slen$ are computed, using the following equations.

$$hlen = \text{round}[tlen(\sigma/2)]$$

(3.6)

where

$$tlen = 2 \text{ round}[slen(\delta/2)].$$

(3.7)

For example, a cepstral window for smoothing the spectrum is shown in Figure 3.9 (a). Figure 3.9 (b) shows the modified group delay spectrum for a frame of speech signal using the customized cepstral smoothing. Another example along with a reference obtained from Wavesurfer (W.S-URL, 2012) is shown in Figure 3.10. In Figure 3.10, pitch peak obtained by the cepstral smoothing is also shown along with formant locations for

\footnote{http://www.phonetik.uni-muenchen.de/studium/}
a frame. Pitch frequency (232 Hz) and formant frequencies (450 Hz, 2540 Hz, 3130 Hz) estimated using the proposed algorithm more or less coincide with the reference computed using Wavesurfer algorithm.

Figure 3.10: MODGD plot for a speech frame. Stems indicate formant locations and pitch. Wavesurfer (W.S) and estimated (Est) values are shown in boxes.

In the proposed algorithm, melodic pitch is computed by mapping the location of the peak in the range \([P_{\text{min}}, P_{\text{max}}]\) in MODGD spectrum to candidate frequency, that ensures consistency. For example, a plot obtained by MODGD analysis for a frame of music is shown in Figure 3.11 (a). The pitch contour obtained by the proposed approach for a music segment is also shown in Figure 3.11 (b). In this approach, the search space in the spectrum is limited by the first formant frequency. For high pitched voices, this results in erroneous pitch estimate where a low formant can be confused for the pitch frequency. Thus the range constraint sets a limit on the performance of the system.

### 3.8.2 Performance evaluation

The performance of the proposed algorithm is evaluated using ADC-2004 dataset, MIREX-2008 dataset, Carnatic music dataset, LabROSA Training set and MIR-1K subset. The evaluation is performed using the metrics RPA and standard deviation of the pitch detection \((\sigma_e)\).

The details of datasets are given below.

- ADC 2004 Dataset (Joo et al., 2010): 20 audio clips (PCM, 16 bit, 44.1 kHz) consisting of daisy, jazz, opera, MIDI and pop genres. Four excerpts are available for every category, each roughly 20 seconds in duration. The corresponding reference data is given at 5.8 ms steps.
Figure 3.11: (a) MODGD plot for a frame of a music sample, (b) Melody pitch extracted using MODGD (Direct).

- LabROSA training set (Tachibana et al., 2010): A referential dataset of MIREX is provided by LabROSA of Columbia University. The data set contains 13 audio files and ground truth $f_0$ data for each audio file. The audio files are of CD-quality (PCM, 16 bit, 44.1 kHz), monaural, 20-30 s length short clips.

- MIREX-2008 (Indian Database) (Ramakrishnan et al., 2008): 4 excerpts of 1 min duration each from North Indian classical vocal performances. There are two different mixtures of each of the 4 excerpts with different accompaniments corresponding to a total of 8 audio clips. These performances consist of the voice, tabla (pitched percussions), tanpura (Indian instrument, perpetual background drone) and a secondary melodic instrument called the harmonium, very similar to the accordion.

- Carnatic: Carnatic music is a sub-genre of Indian classical music. Carnatic music usually, consists of a principal performer (mostly a vocalist), a melodic accompaniment (a violin), a rhythm accompaniment (a mridangam), and a tambura, that acts as a drone throughout the performance. Fourteen Carnatic ālāpanas are used for evaluation purpose. Usually a vocal ālāpana consists of a singer with violin accompaniment all tuned to a specific tonic.

- MIR-1K (Hsu et al., 2012): One thousand song clips with the music accompaniment and the singing voice are recorded at left and right channels, respectively. The duration of each clip ranges from 4 to 13 seconds, and the total length of the
dataset is 133 minutes. These clips are extracted from 110 karaoke songs which contain a mixture track and a music accompaniment track. These songs are freely selected from 5000 Chinese pop songs and sung by 8 females and 11 males. Most of the singers are amateurs and do not have professional music training. Manual annotations of the dataset include pitch contours in semitone, indices and types for unvoiced frames, lyrics, and vocal/non-vocal segments.

In MIR-1K dataset, the music accompaniment and the singing voice are available in left and right channels separately. Singing voice track and accompaniment track are mixed with two signal accompaniment ratio (0 dB and 5 dB). 0 dB means the accompaniment is as strong as the singing voice. The performance of the proposed system is compared with that of YIN algorithm (Cheveigne and Kawahara, 2002.). The results are tabulated in Table 3.2 and Table 3.3. Raw pitch accuracy of 62.80%, 49.00%, 49.07% and 51.00% are reported for ADC, LabROSA, MIREX-2008, Carnatic dataset, respectively. In terms of RPA, the performance of the proposed system is far better than YIN algorithm for ADC, MIREX-2008 dataset but poorer in the case of Carnatic, LabROSA and MIR-1K dataset. The standard deviation error is high as compared to YIN algorithm. In the experiments performed, it is observed that the choice of range \([P_{\text{min}}, P_{\text{max}}]\) plays an important role in the performance of the system. Although pitch estimation is accurate, the range is restrictive. It is observed that the proposed algorithm fails to estimate pitch accurately when the pitch of the voice is high. To overcome such stringent constraints, an alternative method based on the flattened spectrum is proposed in the next section.

![Table 3.2: Comparison of \(\sigma_e\) and RPA](image)

<table>
<thead>
<tr>
<th>Method</th>
<th>RPA (\sigma_e)</th>
<th>ADC (\sigma_e)</th>
<th>LabROSA (\sigma_e)</th>
<th>MIREX-2008 (\sigma_e)</th>
<th>Carnatic (\sigma_e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YIN</td>
<td></td>
<td>50.30</td>
<td>54.81</td>
<td>43.49</td>
<td>64.21</td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td></td>
<td>62.80</td>
<td>49.00</td>
<td>49.07</td>
<td>51.00</td>
</tr>
</tbody>
</table>

![Table 3.3: Comparison of RPA and \(\sigma_e\) for MIR-1K dataset.](image)

<table>
<thead>
<tr>
<th>Method</th>
<th>RPA (\sigma_e)</th>
<th>ADC (\sigma_e)</th>
<th>LabROSA (\sigma_e)</th>
<th>MIREX-2008 (\sigma_e)</th>
<th>Carnatic (\sigma_e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YIN</td>
<td></td>
<td>50.30</td>
<td>54.81</td>
<td>43.49</td>
<td>64.21</td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td></td>
<td>62.80</td>
<td>49.00</td>
<td>49.07</td>
<td>51.00</td>
</tr>
</tbody>
</table>
3.9 Melody pitch extraction using flattened spectrum

As discussed in Section 3.8, the proposed method results in erroneous pitch estimates when the pitch is very close to the first formant. Since we are interested in source information only, if the system information can be annihilated, and the residual signal is subjected to MODGD analysis there is a possibility of obtaining better pitch estimates. This is precisely what is attempted next. Modified group delay analysis on flattened spectrum is denoted as MODGD (Source) and it is discussed below.

3.9.1 Monopitch estimation using MODGD (Source)

Pitch estimation in speech using modified group delay was first attempted in (Murthy, 1991). The power spectrum of the speech is flattened and the modified power spectrum is analysed using MODGD algorithm. The process is illustrated in Figure 3.12. Figure 3.12 (a) shows a frame of speech. Figure 3.12 (b) and Figure 3.12 (c) represent flattened spectrum and MODGD obtained on the flattened spectrum, respectively. Pitch estimated for an entire speech utterance using MODGD (Source) is shown in Figure 3.12 (d) along with the references. The performance is at par with any of the state-of-the-art algorithm.

![Figure 3.12](image_url)

Figure 3.12: (a) Frame of a speech, (b) Flattened power spectrum, (c) Peaks in the MODGD feature space, (d) Pitch estimated for the entire utterance with reference.
3.9.2 Melody pitch extraction using MODGD (Source)

In polyphonic music, strong overlap between partials of different instruments and vocal is present. The underlying task is to estimate the predominant pitch sequence. A schematic diagram of the proposed algorithm is presented in Figure 3.13. The power spectrum is flattened by annihilating the spectral magnitude contribution of the system (filter) characteristics using root cepstrum based smoothing. Root cepstral smoothing is employed, as the objective is to capture the dynamic range of the signal accurately, while at the same time not emphasising window artifacts. The process of spectral flattening for a music frame is illustrated in Figure 3.14. A music frame is shown in Figure 3.14 (a). It’s magnitude spectrum and spectral envelope are shown in Figure 3.14 (b) and (c), respectively. The picket fence harmonics in the flattened spectrum are clearly shown in 3.14 (d).

In the proposed method, flattened spectrum is analysed using MODGD processing. Adaptive windowing is used to capture the dynamic variation of melody. The autocorrelation function is used to determine the transient segments and a window size is chosen accordingly. Salient peaks at multiples of pitch periods can be found in the MODGD feature space. A block by block consistency check on the peak locations helps to select the track corresponding to the melodic pitch using dynamic programming. In the post-processing stage, spurious pitch estimates are removed using median smoothing. A voice activity detection based on harmonic energy is implemented to identify the melodic segments. Assuming a source-system model for production of sounds in mu-
sic, melody in music corresponds to the periodicity of the predominant source. If the timbral information can be suppressed, the picket fence harmonics are essentially pure sinusoids (see Figure 3.14). Owing to the artifacts introduced by windowing, the sinusoids can be thought of as sinusoids in noise. Consider the $z$-transform of a monopitch stream of impulses separated by $T_0$. Then,

$$E(z) = 1 + z^{-T_0} + z^{-2T_0} + ... + ...z^{-nT_0} + ...$$ (3.8)

Assume that a frame contains at most three pitch period. The power spectrum of the source is given by

$$E(z)E^*(z) = (1 + z^{-T_0} + z^{-2T_0} + z^{-3T_0})(1 + z^{T_0} + z^{2T_0} + z^{3T_0}).$$ (3.9)

Substituting $z = e^{j\omega}$,

$$|E(e^{j\omega})|^2 = (1 + \cos(\omega T_0) + \cos(2\omega T_0) + \cos(3\omega T_0))^2 + (\sin(\omega T_0) + \sin(2\omega T_0) + \sin(3\omega T_0))^2$$ (3.10)

$$|E(e^{j\omega})|^2 = 4 + 6 \cos(\omega T_0) + 4 \cos(2\omega T_0) + 2 \cos(3\omega T_0).$$ (3.11)
The root power spectrum is given by:

\[ |E(e^{j\omega})|^{2\gamma} = |4 + 6 \cos(\omega T_0) + 4 \cos(\omega 2T_0) + 2 \cos(\omega 3T_0)|^{\gamma} \]  

(3.12)

where \(0 < \gamma \leq 1\). The parameter \(\gamma\) controls the flatness of the spectrum. Restricting to two impulses per frame and evaluating the power spectrum on the unit circle, we have

\[ |E(e^{j\omega})|^{2\gamma} = 2\gamma (1 + \cos(\omega T_0))^{\gamma}. \]  

(3.13)

Clearly from the signal in Equation 3.12, it can be seen that the flattened power spectrum becomes a sum of sinusoids that are integral multiples of \(\frac{1}{T_0}\). When the dc component is removed, the signal is similar to that of a noisy sinusoid. The task is to estimate the frequency of the sinusoid in this signal. Replacing \(\omega\) by \(n\) and \(T_0\) by \(\omega_0\) in Equation 3.12 yields:

\[ s[n] = a \cos(n\omega_0) + b \cos(n2\omega_0) + c \cos(n3\omega_0) + \ldots, \]  

(3.14)

where \(a, b, c\) are constants. This signal is subjected to modified group delay processing to obtain pitch salience function, which is used in subsequent steps to estimate melodic pitch.

When the signal is polyphonic, \(E(z)\) becomes:

\[
E(z) = 1 + z^{-T_0} + z^{-2T_0} + \ldots + z^{-nT_0} + \\
z^{-T_1} + z^{-2T_1} + \ldots + z^{-nT_1} + \\
z^{-T_2} + z^{-2T_2} + \ldots + z^{-nT_2} + \\
+ z^{-T_i} + z^{-2T_i} + \ldots + z^{-nT_i} + \ldots
\]  

(3.15)

where \(T_i\) corresponds to the pitch of each of the sources. Assuming at most two pitch components per frame, in the group delay domain, additional peaks at \(nT_i - mT_j\) are introduced, where \(T_i\) and \(T_j\) correspond to the pitches of two different sources and \(n, m\) are integers. Nevertheless, the salience corresponding to the prominent pitch is high. The additional pitch estimates due to polyphony are eliminated in dynamic programming based post processing. The exponential decay of the group delay function
at each of these composite pitches is still observed.

3.9.3 Transient analysis using a Multi-resolution Framework

An adaptive window to capture the dynamic variation of melody (Joo et al., 2010) instead of a fixed window (Poliner et al., 2007) is used in the proposed method. Common instances include abrupt changes in amplitude or frequency because of the variation in energy input of the performer, attack regions followed by onsets of percussive sources, fast transitions in frequencies due to expressive pitch variations, and noise due to environmental sounds (Thornburg, 2005). If the input signal changes pitch during an analysis frame or if there is a significant transient, the resulting pitch measurement may be wrong. The duration of the short analysis window does play an important role in accurate pitch marking. A longer window provides high frequency resolution but leads to poor temporal resolution. On the other hand, the use of large windows fails to pick up the rapid pitch modulation that may occur during the transients. Thus, the proposed algorithm uses a multi-resolution framework in which short windows are used for transient segments and long windows otherwise. Signal-driven window length adaptation has been used extensively in audio coding algorithms for capturing the information in stationary and transient audio segments (Painter and Spanias, 2000; Jones and Parks, 1990). Signal sparsity indicators such as Kurtosis measure, L2 norm, Gini index and Spectral flatness are widely used to automatically adapt the window length (Rao et al., 2012). The proposed algorithm uses correlation coefficient (Ref: Equation 3.16) computed on consecutive frames at a lag ($\tau$), to switch between two windows. At zero lag, the autocorrelation coefficient will be the largest. The choice of $\tau$ decides whether the frame is a transient or not. A low autocorrelation coefficient value indicates a transient segment and the window size is resized to a smaller length in such a case. The autocorrelation coefficient $\rho(X, \tau, l)$ of a signal $X$ at the $l^{th}$ frame is defined in (Joo et al., 2010) as,

$$\rho(X, \tau, l) = \frac{\sum_k |X(k; l)| |X(k; l + \tau)|}{\sqrt{\sum_k |X(k; l)|^2 |X(k; l + \tau)|^2}}$$  \hspace{1cm} (3.16)

where, $X(k; l)$ denotes the $k^{th}$ coefficient of the discrete Fourier transform of the $l^{th}$ frame. $\tau$ corresponds to the autocorrelation lag. Discrete Fourier transform of two frames at a lag ($\tau$) of 50 ms, has been empirically chosen for the experiments. Two different window functions are chosen, one that is short and other that is long. In the
proposed algorithm, a threshold of 0.98 on the autocorrelation coefficient is used as a measure to switch between the two windows.

### 3.9.4 Pitch tracking using dynamic programming

The peak locations in the modified group delay functions are the possible pitch candidates in each frame. Dynamic programming is then applied to these peaks to obtain the optimal sequence of pitch candidates across frames. Dynamic programming, where pitch salience is defined by the height of the group delay peak in tandem with transition cost between successive frames is used for this purpose. Thus dynamic programming combines two sources of information for pitch period marking. One source of information is the ‘local’ information corresponding to amplitudes of peaks in the MODGD feature space. The other source of information is the ‘transition’ information corresponding to the relative closeness of the distance between peaks in two consecutive frames in MODGD feature space. The optimal path is selected by minimizing the total cost, which is the sum of local cost and transition cost.

![Figure 3.15: Computation of optimal path by dynamic programming](image)

The computation of local cost, transition cost and optimal path selection are discussed next (Veldhuis, 2000). Local cost function $C_l(c)$ (pitch salience) of a pitch candidate $c$ in the MODGD feature domain is computed by

$$C_l(c) = 1 - \frac{F(c)}{F_{max}}$$  \hspace{1cm} (3.17)
where $F(c)$ is the value of the peak of the pitch candidate $c$ and $F_{\text{max}}$ is the maximum value of the peak present in MODGD feature space for each frame. The transition cost $C_t(c_j/c_{j-1})$ is the distance between the pitch candidates $c_j$ and $c_{j-1}$ of consecutive frames and is defined as

$$C_t(c_j/c_{j-1}) = \frac{|L_j - L_{j-1}|}{l_{\text{max}}} \tag{3.18}$$

where $L_j, L_{j-1}$ are peak locations in consecutive frames. $l_{\text{max}}$ is the maximum transition distance between peaks, computed from the range $[P_{\text{min}}, P_{\text{max}}]$. The transition cost is normalized with the maximum possible transition distance (empirically chosen).

The dynamic programming algorithm finds an optimal pitch sequence $(c_1...c_M)$ with candidates $c_1$ in the first and $c_M$ in the $M^{th}$ frame in a block by minimizing the total cost (TC) (Veldhuis, 2000).

$$\text{Total cost } (TC) = \text{Local Cost} + \text{Transition Cost}. \tag{3.19}$$

Total cost $TC(c_1...c_M)$ of pitch candidates $c_1$ to $c_M$ is computed by

$$TC(c_1...c_M) = C_l(c_1) + \sum_{j=2}^{j=M} C_t(c_j/c_{j-1}) + C_l(c_j). \tag{3.20}$$

The optimal sequence of pitch markers is determined by back tracking from the candidate $c_M$ in the $M^{th}$ frame in a block to its starting frame while minimizing the total cost.

Assume that $c, d$ are the pitch candidates in two adjacent frames. Let $\{e_1,e_2,...,e_k\}$ be the successors of a candidate $d$ for $k$ frames. The minimum distance from the candidate $d$ to its successors, $C_{\text{min}}(d)$ is computed by minimizing Equation 3.20 and is denoted as,

$$C_{\text{min}}(d) = \min_{e_1,...,e_k} C(e_1,...,e_k,d). \tag{3.21}$$

The optimal pitch sequence starting from candidate $c$ followed by $d$ in the consecutive frame, with successors $\{e_1,e_2,...,e_k\}$ is computed as

$$TC_{\text{min}} = C_l(c) + \min(C_{\text{min}}(d) + C_t(c/d)). \tag{3.22}$$
Figure 3.15 illustrates the process of finding the optimal path for a block of consecutive frames. Peak locations for four consecutive frames are shown in the left block and corresponding pitch saliences are shown in the right block. Peak locations are the positions of peaks in the MODGD feature space with a magnitude above a threshold. Pitch salience is the corresponding local cost computed as defined in Equation 3.17. Cost estimation starts from the last frame in the block i.e \{73, 128, 178, 232\}, with corresponding local cost \{0.88, 0.85, 0.92, 0.89\} and backtracks until the first frame of the block. After estimating the cost for all possible paths, the path with minimum total cost is chosen as the optimal path. Optimal path computed by minimizing the total cost is highlighted in the block obtained after dynamic programming. The optimal path chosen in the given example is \{129, 124, 133, 128\} highlighted in box. Block by block dynamic programming across sequence of voiced frames results in the optimal path of the pitch sequence by ensuring continuity.

The optimal path obtained from the dynamic programming stage is selected and a search is performed over the MODGD feature space at multiples of the first optimal path. Figure 3.16 (a) shows the peaks at multiples of pitch for a music frame. This location information is also used to reinforce the value of pitch obtained.

![Figure 3.16: (a) MODGD plot for a frame, (b) Melody pitch extraction for opera fem4.wav using MODGD (Source).](image)
Pitch halving/doubling is reduced considerably by context-framed rules. Instantaneous transitions in the intra-voiced regions are not permitted. If detected, it is replaced by an interpolated sequence followed by a median smoothing to ensure pitch continuity through a post processing stage. Median smoothing is performed to eliminate spurious pitch estimates occurred in the process of pitch computation. The predominant pitch extracted using the proposed method is compared with ground truth for audio of a female singer opera fem4.wav in Figure 3.16 (b). Figures 3.17 (a) and 3.17 (b) show plots of the melodic pitch contour for a North Indian classical excerpt and a Carnatic music excerpt, respectively.

### 3.9.5 Voice Activity Detection (VAD)

Most melody extraction algorithms also include voicing detection using loudness threshold (Brossier, 2005), adaptive threshold (Cancela, 2008) and harmonic energy (Ramakrishnan et al., 2008) as criteria. HMM based detector with a 2-state HMM to decode the mixture input into voiced and unvoiced segments is used in (Hsu et al., 2009). As most approaches yield more or less the same performance, we have used the normalized harmonic energy for voice activity detection. The frame-wise normalized harmonic energy is sorted in ascending order and the value at 25% of size of the array is set as the threshold. The harmonic energy is computed from the estimated fundamental frequency $f_0$ and its harmonics up to 5 kHz (Ramakrishnan et al., 2008). Multiples of the fundamental frequency are identified by searching for the local maxima with 3% tolerance.
3.9.6 Performance Evaluation

The proposed algorithm is evaluated using four datasets. The datasets used in this evaluation are given in Table 3.4. The duration referred to in Table 3.4 corresponds to the total duration of all the audio clips in the given database. All the recordings in the LabROSA, MIREX-2008 and Carnatic music datasets are processed frame-wise with a hop size of 10 ms while ADC-2004 dataset is processed with a hop size of 5.8 ms (reference in the dataset is given for hop size of 5.8 ms).

Table 3.4: Description of the datasets used for the evaluation

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Description</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC-2004</td>
<td>MIREX-2004 dataset (5 styles)</td>
<td>368sec</td>
</tr>
<tr>
<td>LabROSA</td>
<td>MIREX-05 Training set (13 Files)</td>
<td>391sec</td>
</tr>
<tr>
<td>MIREX-2008</td>
<td>North Indian Classical Music (8 Files)</td>
<td>500sec</td>
</tr>
<tr>
<td>CDB</td>
<td>Carnatic Music (14 Files)</td>
<td>876sec</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>2135sec</strong></td>
</tr>
</tbody>
</table>

Evaluation methodology is same as that of MIREX standards. The performance of
the proposed algorithm is evaluated using six metrics discussed in Section 3.5.

3.9.7 Results, analysis and discussion

Table 3.5 compares the performance of the proposed algorithms with various methods submitted by the participants for MIREX evaluation in 2011, 2012, 2013 and 2014 on the ADC-2004 dataset. Table 3.6 tabulates the metrics for the MIREX-2008 dataset. Most of the techniques in Table 3.5 and Table 3.6 are magnitude spectrum based approaches. The proposed MODGD (Source) phase based method performs at par with the performance of other magnitude spectrum based approaches. As far as the performance is concerned, RPA of the proposed algorithm appears to be good for styles where the singing melody is predominant. In the ADC-2004 dataset, singing melody is predominant in daisy, opera and pop. From the box plot in Figure 3.18, it is obvious that raw pitch accuracy is higher for those styles as compared to others.

![Figure 3.18: Raw pitch accuracy of ADC-2004 dataset in box plot](image)

Adaptive windowing is incorporated in the proposed algorithm to capture dynamic variation in the melody. The experiment is performed on pop music with various fixed and adapted window sizes. Experimental results on pop music genre, selected from ADC-2004 dataset using adaptive windowing are illustrated in Figure 3.19. Initially, three fixed window sizes 25 ms, 30 ms, 35 ms are selected and predominant melodic pitch is computed. Later, the experiments are repeated by switching between two windows, one with 80% of previously fixed window based on autocorrelation coefficient criteria. The conjecture is that the smaller window should be used by the transient segments and larger window by the steady state segments. It is obvious from the plot that
the window adaptation has benefited the overall performance. The improved results motivated us to apply this technique to all the datasets.

![Graph showing pitch accuracy vs. window size](image)

Figure 3.19: Results on adaptive windowing on three window sizes

In (Salamon et al., 2014b), it is stated that the best performing algorithms obtain a raw pitch accuracy between 70 and 80%. One of the submissions of Bin Liao et al. outperforms other two submissions due to the effective utilization of three pitch tracking schemes: two trend-estimation-based methods and one HMM-based method (Tzu et al., 2012). Dynamic programming by the pitch salience peak, in-addition to the transition information and adaptive windowing to capture the transients, have effectively contributed to improve the accuracy compared to the first approach (MODGD (Direct)). As compared to the earlier work, imposing rules in the post processing phase also effectively minimized spurious pitch estimates in the melodic pitch computation. From Tables 3.5 and 3.6, it can be seen that the Dressler’s algorithm outperforms most of the algorithms. The algorithm implements an auditory streaming model which builds upon tone objects and salient pitches. The formation of voices is based on the regular update of the frequency and the magnitude of so called streaming agents, which aim at salient tones or pitches close to their preferred frequency range (Dressler, 2011). Dressler’s method gives the best performance for the audio material, which contain a mixture of vocal and instrumental pieces, but performs poorly for the other collections where the melody is always vocal (Salamon et al., 2014b). In the case of MIREX-2008 dataset, MODGD (Source) algorithm achieves almost the same performance as that of PolyPDA algorithm (Rao and Rao, 2010). The results reported for this dataset are 73.9% and 76.3% for RPA and RCA, respectively. The proposed MODGD (Source) algorithm outperforms the algorithm proposed by Stacy Hsueh et al. in RPA and at par with it in OA.
The metrics RPA, RCA and standard deviation of pitch detection error are used for the performance evaluation of LabROSA dataset. We see that the proposed methods achieve better overall performance than ESPS method (wavesurfer) (W.S-URL, 2012), and MODGD(Source) based approach outperforms the PolyPDA algorithm in terms of RPA. From Table 3.7, we can observe that the performance of the proposed MODGD (Source) is at par with that of MELODIA (Salamon and Gomez, 2012) and the algorithm of V.Arrora et al. (Arora and Behera, 2013). The results on the Carnatic music dataset are summarized in Table 3.8. In Carnatic music dataset, accompaniments are less for ālapanas, which results in better performance for MODGD (Source), as compared to other datasets. Since singing voice is the predominant melody line in Carnatic music dataset, \([P_{\text{min}}, P_{\text{max}}]\) range selected is also less for MODGD (Direct) algorithm. In the case of Carnatic music dataset, since the ground truth is not available, pitch estimated using the algorithm proposed by V.Arrora et al. (Arora and Behera, 2013) is used as the reference for objective comparison. It was the best performing system in MIREX-2012 evaluation. The standard deviation of pitch detection error is less as compared to other schemes and in terms of RPA, the proposed algorithm is ranked second. Correctness of the extracted pitch is validated by a professional musician after synthesizing the music from the extracted pitch.

The performance metric, overall accuracy points towards the need for proper choice of voicing step. For example, as seen in the Table 3.5, even though the proposed MODGD (Source) approach is successful in estimating melodic pitches for the majority of frames correctly as that of Salamon et al. (Salamon and Gomez, 2012), voicing detection deteriorated the overall performance. It can be seen that both the proposed approach and Salamon et al.’s algorithm have a similar performance in terms of RPA and RCA for ADC-2004 dataset. Salamon uses a threshold based on the salience distribution of pitch contours to remove non-salient contours before proceeding to filter out other non-melody contours. But, the system performance is sensitive to a parameter which determines the lenience of the filtering. Effective use of auditory approach to separate melody segments resulted in better OA for the algorithm proposed by Dressler (Ref: Table 3.5) (Dressler, 2011). Since, till date most approaches focus primarily on the raw pitch accuracy, and less on the voicing decision, overall algorithmic performance of the melody extraction task will be affected adversely (Salamon et al., 2014b). In voicing detection phase, the trade-off between VF and VR is also considered in fixing
the optimal threshold on harmonic energy. As far as the post-processing step is concerned, while the proposed algorithm relies on the basic post processing techniques, there are algorithms which use different approaches like hidden Markov model (Kum et al., 2016) and group characteristics (Yoon et al., 2011). The performance of the proposed approach actually demonstrates the potential of the phase based method in melody extraction task and other music retrieval applications.

Table 3.5: Comparison of results for ADC-2004 submitted in 2011/2012/2013/2014 evaluation.

<table>
<thead>
<tr>
<th>Method</th>
<th>OA</th>
<th>RPA</th>
<th>RCA</th>
<th>VR</th>
<th>VF</th>
</tr>
</thead>
<tbody>
<tr>
<td>V. Arora et al</td>
<td>69.06</td>
<td>81.41</td>
<td>85.92</td>
<td>76.51</td>
<td>23.56</td>
</tr>
<tr>
<td>Sam Meyer</td>
<td>60.34</td>
<td>64.23</td>
<td>71.21</td>
<td>77.36</td>
<td>32.96</td>
</tr>
<tr>
<td>Bin Liao et al (1)</td>
<td>46.24</td>
<td>55.87</td>
<td>66.71</td>
<td>99.98</td>
<td>97.76</td>
</tr>
<tr>
<td>Bin Liao et al (2)</td>
<td>41.54</td>
<td>48.32</td>
<td>59.90</td>
<td>99.96</td>
<td>95.37</td>
</tr>
<tr>
<td>Bin Liao et al (3)</td>
<td>41.54</td>
<td>48.32</td>
<td>59.90</td>
<td>99.96</td>
<td>95.37</td>
</tr>
<tr>
<td>Salamon et al (1)</td>
<td>73.55</td>
<td>76.34</td>
<td>78.71</td>
<td>80.55</td>
<td>15.09</td>
</tr>
<tr>
<td>Salamon et al (2)</td>
<td>73.97</td>
<td>77.28</td>
<td>79.41</td>
<td>80.64</td>
<td>15.25</td>
</tr>
<tr>
<td>Tachibana et al</td>
<td>59.42</td>
<td>73.03</td>
<td>81.43</td>
<td>74.98</td>
<td>29.37</td>
</tr>
<tr>
<td>Yeh</td>
<td>46.99</td>
<td>56.40</td>
<td>65.33</td>
<td>99.91</td>
<td>93.69</td>
</tr>
<tr>
<td>C Cannam et al</td>
<td>66.69</td>
<td>74.79</td>
<td>78.92</td>
<td>63.63</td>
<td>10.35</td>
</tr>
<tr>
<td>K. Dressler</td>
<td>86.30</td>
<td>87.10</td>
<td>87.62</td>
<td>91.63</td>
<td>15.76</td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td>55.20</td>
<td>62.80</td>
<td>76.03</td>
<td>80.26</td>
<td>55.00</td>
</tr>
<tr>
<td>MODGD (Source)</td>
<td>65.38</td>
<td>72.01</td>
<td>76.95</td>
<td>82.26</td>
<td>26.00</td>
</tr>
</tbody>
</table>

Table 3.6: Comparison of results for MIREX-2008 submitted in 2011/2012/2013/2014 evaluation.

<table>
<thead>
<tr>
<th>Method</th>
<th>OA</th>
<th>RPA</th>
<th>RCA</th>
<th>VR</th>
<th>VF</th>
</tr>
</thead>
<tbody>
<tr>
<td>V. Arora et al</td>
<td>67.95</td>
<td>85.85</td>
<td>86.79</td>
<td>70.76</td>
<td>15.58</td>
</tr>
<tr>
<td>Sam Meyer</td>
<td>50.06</td>
<td>49.31</td>
<td>59.48</td>
<td>63.52</td>
<td>30.23</td>
</tr>
<tr>
<td>Bin Liao et al (1)</td>
<td>70.25</td>
<td>81.94</td>
<td>82.17</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Bin Liao et al (2)</td>
<td>51.21</td>
<td>59.59</td>
<td>67.95</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Bin Liao et al (3)</td>
<td>51.51</td>
<td>59.59</td>
<td>67.95</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Salamon et al</td>
<td>82.78</td>
<td>87.55</td>
<td>88.02</td>
<td>89.26</td>
<td>17.86</td>
</tr>
<tr>
<td>Stacy Hsu et al</td>
<td>63.57</td>
<td>67.64</td>
<td>73.20</td>
<td>78.69</td>
<td>34.25</td>
</tr>
<tr>
<td>C Cannam et al</td>
<td>57.67</td>
<td>73.35</td>
<td>74.43</td>
<td>68.21</td>
<td>44.64</td>
</tr>
<tr>
<td>K. Dressler</td>
<td>86.30</td>
<td>87.73</td>
<td>88.48</td>
<td>91.26</td>
<td>28.58</td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td>44.01</td>
<td>49.00</td>
<td>60.05</td>
<td>80.30</td>
<td>34.96</td>
</tr>
<tr>
<td>MODGD (Source)</td>
<td>63.01</td>
<td>71.06</td>
<td>73.06</td>
<td>83.55</td>
<td>28.27</td>
</tr>
</tbody>
</table>

Table 3.7: Comparison of $\sigma_e$, RPA, RCA for LabROSA training dataset

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Method</th>
<th>$\sigma_e$</th>
<th>RPA (%)</th>
<th>RCA (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESPS method (Wavesurfer)</td>
<td>2.97</td>
<td>25.52</td>
<td>62.94</td>
<td></td>
</tr>
<tr>
<td>PolyPDA</td>
<td>1.70</td>
<td>62.87</td>
<td>72.67</td>
<td></td>
</tr>
<tr>
<td>V. Arora et al</td>
<td>1.85</td>
<td>66.00</td>
<td>72.80</td>
<td></td>
</tr>
<tr>
<td>Salamon et al (MELODIA)</td>
<td>2.10</td>
<td>68.62</td>
<td>71.26</td>
<td></td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td>3.37</td>
<td>49.07</td>
<td>60.05</td>
<td></td>
</tr>
<tr>
<td>MODGD (Source)</td>
<td>2.67</td>
<td>64.01</td>
<td>69.54</td>
<td></td>
</tr>
</tbody>
</table>
### Table 3.8: Comparison of $\sigma_e$, $RPA$, $RCA$ for *Carnatic* music dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>$\sigma_e$</th>
<th>$RPA$ (%)</th>
<th>$RCA$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MELODIA</td>
<td>2.62</td>
<td>85.21</td>
<td>90.70</td>
</tr>
<tr>
<td>PolyPDA</td>
<td>2.41</td>
<td>89.20</td>
<td>92.03</td>
</tr>
<tr>
<td>MODGD (Direct)</td>
<td>3.43</td>
<td>51.00</td>
<td>70.48</td>
</tr>
<tr>
<td>MODGD (Source)</td>
<td>2.30</td>
<td>85.50</td>
<td>88.49</td>
</tr>
</tbody>
</table>

**Performance analysis on varying degree of polyphony**

As mentioned in (Salamon *et al.*, 2014b), varying degree of polyphony is a challenging issue in the task of audio melody extraction. Regardless of the lead melody line, the task becomes harder as we increase the number of instruments in orchestral background. In such instances, the pitch contour of the predominant melodic instrument may be closer to melody line of other accompanying instruments, which makes the task of melody extraction more complex (Salamon *et al.*, 2014b). Salamon *et al.* discuss the effect of degree of polyphony in (Salamon *et al.*, 2015).

![Figure 3.20](image-url)

Figure 3.20: (a) RPA and RCA for DB1 [less accompaniments], (b) RPA and RCA for DB2 [5-6 accompaniments], (W.S : Wavesurfer, pYIN : Probabilistic YIN, MEL : MELODIA, MODGD : MODGD (Source))

The proposed MODGD (Source) algorithm is also evaluated on varying level of polyphony using MedleyDB dataset (Bittner *et al.*, 2014). This dataset consists of annotated, royalty free multitrack recordings. The dataset consists of 122 songs, 108 of which include melody annotations. The dataset consists of a stereo mix and both dry
and processed multitrack stems for each song. During recording process, a set of microphones is used, such that there may be more than one microphone recording a single source. The resulting files are raw unprocessed mono audio tracks. The raw files are then grouped into stems, each corresponding to a specific sound source. The experiments are conducted using two customized datasets; 1) leading singing voice with 1-2 accompaniments (DB1), 2) leading singing voice with 5-6 accompaniments of wider pitch ranges (DB2). DB1 and DB2 are created by mixing leading singing voices with background music stems available in the dataset using Audacity. The performance of the melody extraction task on DB1, DB2 is evaluated using Wavesurfer (W.S-URL, 2012), pYIN (Mauch and Dixon, 2014), MELODIA and MODGD (Source) algorithms. The trend in the results can be seen in Figure 3.20 (a) and (b). As the number of accompaniments increases, considerable decrease in pitch accuracy is reported for Wavesurfer and pYIN. The proposed MODGD (Source) algorithm and MELODIA yield a competitive performance, both with RCA greater than 80% on DB1. As the task gets tougher moving on from DB1 to DB2, where the degree of polyphony is high, the performance of the algorithms declined but less as compared to pYIN and Wavesurfer. Even in the high degree of polyphony, chroma accuracy of 70% is promising in melody extraction context.

3.10 Automatic music genre classification

Automatic musical genre classification provides a framework for developing and evaluating features for content-based analysis of music signals. Due to incomplete and inconsistent metadata associated with a large collection of music files, an increasing demand for automatic retrieval of music content exists nowadays. Automatic music genre classification task extracts distinctive features from audio excerpts and then automatically categorises them into musical genres. A genre or sub-genre can be distinguished from each other by the style, the cultural context or the content and spirit of the themes. Although the division of music into genres is somewhat subjective and arbitrary, there exists perceptual criteria based on the texture, instrumentation and rhythmic structure of music that can be used to characterize a particular genre (Tzanetakis et al., 2001). Automatic music genre classification can be utilized effectively to automatically select radio stations playing a particular genre of music or to design a selective band equalizer based on the label (Barbedo and Lopes, 2007).
A comprehensive survey of both features and classification techniques used in genre classification can be found in (N.Scaringella et al., 2006). In (Salamon et al., 2012b), melodic features are computed directly from the polyphonic music and features are fused at feature level with MFCC features to develop a genre classification scheme. In (Aryafar et al., 2012), a method is introduced with 1-SVM classifier which combines the ideas of the classical SVM with the sparse approximation techniques in genre classification. A similar approach is also proposed in (Xu et al., 2003). Riedmiller et al. use unsupervised learning to create a dictionary of features (Wulfing and Riedmille, 2012) as part of the task. Two novel classifiers, using inter-genre similarity (IGS) modeling are proposed in (UlaBagci and Erzin, 2007). The method also investigates the use of dynamic timbrel texture features in order to improve automatic musical genre classification performance. In (Tzanetakis and Cook, 2002), three feature sets for representing timbrel texture, rhythmic content and pitch content are proposed. Extraction of psychoacoustic features related to music surface and high level semantic descriptions are explored in (Scheirer, 2000) for music genre classification. The proposed method in this work uses fusion of melodic features and modified group delay features for automatic music genre classification task.

### 3.10.1 Proposed method

The proposed system is shown in Figure 3.21. The algorithm uses fusion of high level features derived from melodic contour with low level features to effectively map the significant characteristics of genres. Initially, modified group delay features are used to perform genre classification. Later, modified group delay features are fused with high level features derived from the predominant pitch contour to create a new feature set. Support vector classifier is used in the classification phase. High level melodic features include mean pitch height, pitch deviation, jitter, skewness, kurtosis and bandwidth.
Figure 3.22: Spectrogram (top) and Modgdgram (bottom) for a song from jazz genre derived from the kernel density estimate of melodic pitch sequences. The following section describes the feature extraction and classification in detail.

3.10.2 Modified group delay feature (MODGDF)

In the proposed scheme, modified group delay features are used for genre classification. 13 dimensional MODGDF features computed framewise are used for fusion experiments. Modified group delay functions effectively map system characteristics in feature space. For example, the modgdgram and spectrogram of two songs, one from blues and the other from jazz are shown in Figure 3.22. It is obvious that modgdgram emphasizes system-specific information when compared to spectrogram.

3.10.3 Melodic Features

Prominent pitch or melodic pitch is the pitch contour of the lead or dominant musical instrument in a song. In music with orchestration, there are several musical instruments playing along with the singing voice which result in strong interferences. MELODIA (Salamon and Gomez, 2012) is used for computing the prominent melodic pitch of all

---

9 Visual representation of modified group delay functions with time in vertical axis and frame index in horizontal axis. A third dimension, indicating the amplitude of group delay function at a particular time is represented by the intensity or color of each point in the image
the excerpts in the proposed system. In (Salamon and Gomez, 2012), pitch candidates are first estimated, and the context is used to determine pitch contours. Peaks of the function are grouped over time using auditory streaming cues into pitch contours: time and frequency continuous sequences of salient pitches. Contour features such as length, height, pitch deviation and presence of vibrato are then used for filtering non-melody segments. A set of contour characteristics is computed for each contour and used to filter out non-melodic contours. Pitch is extracted using a 10 ms hopsize.

The following features are computed from the prominent melodic pitch contour and used in the experiments.

- **Mean pitch height** ($\mu_p$): Mean pitch height is given by,

$$\mu_p = \frac{1}{N} \sum_{i=1}^{N} P_i$$

(3.23)

where $P_i, N$ represent pitch for a frame, number of frames, respectively.

- **Pitch deviation** ($P_{dev}$): Pitch deviation is computed using the equation:

$$P_{dev} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (P_i - \mu_p)^2}$$

(3.24)

- **Jitter**: It is the cycle-to-cycle variation of fundamental frequency, i.e. the average absolute difference between consecutive periods, expressed as:

$$Jitter(absolute) = \frac{1}{N - 1} \sum_{i=2}^{N} |P_i - P_{i-1}|$$

(3.25)

Three parameters; skewness, kurtosis and bandwidth are also computed from the Gaussian kernel density estimate of melodic pitch sequences. Kernel estimators centre a smooth kernel function at each data point to get smooth density estimate when compared to pitch histograms. If we denote the kernel function as $K$ and its bandwidth by $h$, for $n$ data points, the estimated density at any point $x$ is,

$$f(x) = \frac{1}{n} \sum_{i=1}^{N} K\left(\frac{x - x(i)}{h}\right)$$

(3.26)

where $\int K(t)dt = 1$.

- **Skewness**: Skewness is a measure of the asymmetry of the probability distribution. The skewness of a random variable $X$ is the third standardized moment

---

10http://homepages.inf.ed.ac.uk/rbf../kde.html

108 Figure 3.23: Kernel density estimate of melodic pitch for two genres. (a) Pop, (b) Jazz.

\[ \gamma_1, \text{ defined by,} \]
\[ \gamma_1 = E \left[ \left( \frac{X - \mu}{\sigma} \right)^3 \right] = \frac{\mu_3}{\sigma^3} = \frac{E \left[ (X - \mu)^3 \right]}{(E \left[ (X - \mu)^2 \right])^{3/2}} \]  

(3.27)

where \( \mu \) is the mean, \( \sigma \) is the standard deviation, \( E \) is the expectation operator, and \( \mu_3 \) is the third central moment.

- **Kurtosis**: Kurtosis is a measure of the ‘tailedness’ of the probability distribution of random variable. The kurtosis is the fourth standardized moment, defined by,

\[ \text{Kurt}[X] = \frac{\mu_4}{\sigma^4} = \frac{E[(X - \mu)^4]}{(E[(X - \mu)^2])^2}, \]  

(3.28)

where \( \mu_4 \) is the fourth moment about the mean.

- **Bandwidth**: This is the smoothing parameter, \( h \) in Equation (3.26) that approximates the data optimally, when Gaussian approximation is used.

Kernel density estimate (KDE) of two genres are given in Figure 3.23. KDE of pitch sequences varies from genre to genre. While spiky nature of density estimate is observed in Jazz genre, spread estimate becomes the key characteristic of pop music.

---

12https://en.wikipedia.org/wiki/Kurtosis
Table 3.9: Overall accuracy

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Feature Name</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MFCC</td>
<td>66.03</td>
</tr>
<tr>
<td>2</td>
<td>MFCC+ Melodic</td>
<td>69.00</td>
</tr>
<tr>
<td>3</td>
<td>MODGDF</td>
<td>71.73</td>
</tr>
<tr>
<td>4</td>
<td>MODGDF + Melodic</td>
<td>75.50</td>
</tr>
</tbody>
</table>

Table 3.10: Classification accuracies obtained by existing approaches.

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Reference</th>
<th>Dataset</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Li et al. (Li et al., 2003)</td>
<td>GTZAN</td>
<td>78.50</td>
</tr>
<tr>
<td>2</td>
<td>Tzanetakis et al. (Tzanetakis and Cook, 2002)</td>
<td>GTZAN</td>
<td>61.00</td>
</tr>
<tr>
<td>3</td>
<td>Panagakis et al. (C. Panagiotakis, 2005)</td>
<td>GTZAN</td>
<td>78.20</td>
</tr>
<tr>
<td>4</td>
<td>Holzapfel et al. (Holzapfel and Stylianou, 2008b)</td>
<td>GTZAN</td>
<td>74.00</td>
</tr>
<tr>
<td>5</td>
<td>Salamon et al. (Salamon et al., 2012b)</td>
<td>GTZAN</td>
<td>82.00</td>
</tr>
<tr>
<td>6</td>
<td>Holzapfel et al. (Holzapfel and Stylianou, 2008b)</td>
<td>ISMIR2004</td>
<td>83.50</td>
</tr>
<tr>
<td>7</td>
<td>Panagakis et al. (C. Panagiotakis, 2005)</td>
<td>ISMIR2004</td>
<td>80.95</td>
</tr>
</tbody>
</table>

3.10.4 Evaluation Dataset

The performance of the proposed system is evaluated on a subset of GTZAN dataset (Tzanetakis and Cook, 2002). GTZAN dataset is created by Tzanetakis and Cook and it includes 1000 music excerpts of 30 seconds duration with 100 examples in each of 10 different categories. In the proposed work, five genres: Pop, Blues, Classical, Jazz, Metal are considered. The classical dataset has the following classes: choir, orchestra, piano, string quartet whereas the jazz dataset includes big band, cool, fusion, piano, quartet and swing. The tracks are all 22050 Hz Mono 16-bit audio files in .wav format.

Initially, a baseline system using 13 dimensional MFCC with SVM classifier is created. In the second phase, the experiment is repeated with modified group delay features. MFCC and MODGDF (13 dimensions) feature sets are converted to utterance level by averaging across dimensions and a representative feature vector is created for each utterance. The experiments are extended by combining 6 dimensional melodic features to MFCC/MODGDF features to create a new feature set (19 dimensions). In all the experiments, 60% of files are used for training and the rest for testing.
3.10.5 Results and analysis

The results are tabulated in Table 3.9. Classification accuracies obtained by existing approaches in music genre classification are also summarized in Table 3.10. Since the classification framework, test-train pattern, size of the dataset and segmental length of the test pattern are different for each of the proposed methods in Table 3.10, direct comparison only helps to understand the trend in classification framework. The proposed system shows the effectiveness of the fusion of group delay features and melodic features in automatic music genre classification. Whilst the baseline MFCC system reports overall accuracy of 66.33%, modified group delay based system reports 71.73%. When we combine the high level melodic features with low-level features, significant improvement in performance is noted in both systems. Fusion of melodic features with MFCC and MODGDF achieves an increase of 3% and 4%, respectively. Genre wise classification for all the experiments are shown in Figure 3.24. For all the genres except metal, MODGDF fusion has shown improvement in classification accuracy. The confusion matrix for MODGDF fusion experiment is shown in Table 3.11. 39 excerpts out of 40 pop music files are correctly classified, but for jazz music more misclassification is observed. Playing instruments using a synthesizer and drums are the uniqueness of the pop music, which is effectively mapped by MODGDF/MFCC features. Guitar, base drums and saxophone are common in jazz and pop, which might have lead to more misclassification. Moreover, classical music excerpts which employ only instruments are effectively classified by MODGDF/fusion rather than MFCC alone. It is worth noting that MODGDF based system alone outperforms the MFCC fused system except in the case of blues. Overall, the experiments show the potential of the fusion of melodic
Table 3.11: Confusion matrix of fusion (MODGDF + melodic) experiment.

<table>
<thead>
<tr>
<th></th>
<th>Blues</th>
<th>Classical</th>
<th>Jazz</th>
<th>Metal</th>
<th>Pop</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blues</td>
<td>23</td>
<td>0</td>
<td>10</td>
<td>6</td>
<td>0</td>
<td>58.97</td>
</tr>
<tr>
<td>Classical</td>
<td>1</td>
<td>38</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>95.00</td>
</tr>
<tr>
<td>Jazz</td>
<td>3</td>
<td>4</td>
<td>18</td>
<td>0</td>
<td>15</td>
<td>45.00</td>
</tr>
<tr>
<td>Metal</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>32</td>
<td>0</td>
<td>80.00</td>
</tr>
<tr>
<td>Pop</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>39</td>
<td>97.50</td>
</tr>
</tbody>
</table>

features with group delay features in automatic music genre classification.

**Summary**

Two techniques, both based on MODGDF are proposed for automatically extracting melody from polyphonic music. In the first approach, the music signal is analysed using modified group delay functions and prominent pitch sequences are estimated. But due to stringent constraints on the choice of window function, a method is proposed in which MODGD analysis is performed on the flattened spectrum with system characteristics cancelled. In MODGD (Source) based method, the power spectrum is first flattened to yield a signal that is rich in harmonics. The harmonically rich spectrum is then subjected to modified group delay processing for the estimation of sinusoids in noise. The window size is selected based on adaptive windowing technique using the autocorrelation function to capture the dynamic variation of melody. Dynamic programming is used to ensure consistency across frames. An important feature of the proposed algorithm is that it neither requires any substantial prior knowledge of the structure of musical pitch nor any classification framework. Performance of the algorithms on standard databases do indicate that the proposed algorithm shows promise. The high level melodic features computed from the melodic contour are fused with baseline MFCC/MODGD features to improve the classification accuracy in automatic music genre classification. This is also experimentally validated using GTZAN dataset. Overall, one can conclude that modified group delay based pitch extraction is promising.
CHAPTER 4

Two-pitch Tracking in Co-channel Speech

4.1 Introduction

Detecting predominant pitch/multiple pitches that is/are present in speech or music is a very complex task. In common instances such as in crowded party, meetings, where many people talk simultaneously with varying intonations, it is very difficult to attend to a particular voice and it is referred to as “cocktail problem” (Cherry, 1953). When a combination of speech utterances from two or more speakers are transmitted, pitch cues of the individual sources will be weakened by the presence of mutual interference. In such ambiguous situations, estimating accurate pitch tracks is a challenging task. In this chapter, an algorithm based on modified group delay functions is proposed for two-pitch tracking in co-channel speech. Section 4.2 formulates the problem of multipitch estimation in speech. State-of-the-art approaches are explained in Section 4.3. Section 4.4 describes the evaluation methodology along with metrics and datasets. Few applications of multipitch estimation in speech and music are listed in Section 4.5. The proposed algorithm for two-pitch tracking using modified group delay functions is given in Section 4.6. The performance of the proposed algorithm is evaluated using standard metrics in Section 4.7 followed by results and analysis in Section 4.8. Finally, the promise of the algorithm in multipitch environment is experimentally shown using three-speaker mixed speech in Section 4.9.

4.2 Multipitch Estimation

Multipitch analysis is the task of analyzing the pitch content of polyphonic audio such as multi-talker speech, polyphonic music and multi-bird songs. It includes estimating the frequency and number of pitches at each time frame, and organizing the pitches according to the sources. Several factors make multiple voice $f_0$ estimation more challenging than single voice $f_0$ estimation. Mutual overlap between voices weaken their pitch cues and make it difficult to extract the individual estimates.
According to MIREX, multipitch analysis can be approached at three levels.  

- **Level 1** - Multi-pitch estimation is to collectively estimate pitch values of all concurrent sources at each individual time frame without determining their sources.

- **Level 2** - Note tracking is to estimate continuous pitch segments that typically correspond to individual notes or syllables.

- **Level 3** - Multipitch estimation and streaming is to estimate pitches and stream them into a single pitch trajectory over an entire conversation or music performance for each of the concurrent sources.

Mathematically, the multi-pitch estimation problem can be formulated as follows (Christensen et al., 2008): Consider a signal consisting of several, say \( K \), sets of harmonics with fundamental frequencies \( \omega_k \), for \( k = 1, \ldots, K \), that is corrupted by an additive white Gaussian noise \( \omega[n] \), having variance \( \sigma^2 \), for \( n = 0, \ldots, N - 1 \), i.e.,

\[
x[n] = \sum_{k=1}^{K} \sum_{l=1}^{L} a_{k,l} e^{j\omega_k l n} + \omega[n] \quad (4.1)
\]

where \( a_{k,l} = A_{k,l} e^{j\phi_{k,l}} \) is the complex amplitude of the \( l^{th} \) harmonic of the source with \( A_{k,l} > 0 \), \( \phi_{k,l} \) being the amplitude and the phase of the \( l^{th} \) harmonic of the \( k^{th} \) source, respectively. The model in Equation (4.1) is known as the harmonic sinusoidal model. The task is to estimate the individual pitch estimates \( \omega_k \) in the mixture signal. Most of the pitch detection algorithms are designed to estimate pitch in single speaker cases. Monopitch estimation algorithms fail in estimating multiple pitches in mixed

1http://www.ece.rochester.edu/ zduan/multipitch/
speech. The pitch estimated for a mixed speech utterance, using Wavesurfer algorithm (W.S-URL, 2012) is shown in Figure 4.1. It fails to track individual pitches correctly in the presence of strong interference from another speaker. Multiple $f_0$ estimation systems can be broadly classified according to their mid-level representation (time domain, spectral domain, auditory filter banks) or by the way they can estimate the interactions between sources (iterative and joint estimation methods) (Ibnanez, 2010).

### 4.3 Multipitch estimation algorithms: State-of-the-art approaches

Numerous methods have been reported for multipitch estimation in speech and music (Li et al., 2008; Nishimoto et al., 2007; Wu and Wang, 2003). The research on multipitch estimation was initiated through separating co-channel speech signals date back to 1970 (Shields, 1970). All the approaches can be classified into one of the three classes, spectral, temporal and spectro-temporal. The existing strategies include iterative estimation or joint estimation, in time/frequency domain. In iterative estimation, a single-voice algorithm is applied to estimate the $f_0$ of one voice, and that information is then used to suppress the voice from the mixture so that the fundamental frequencies of the other voices can be estimated. Suppression of the other voices, may in turn be used to refine the estimate of the first voice. In joint estimation, all the voices are estimated at the same time. Whatever be the approach, there exists a compromise between the computational cost and robustness in the performance. Joint estimation does the source separation better than iterative estimation but with high computational complexity (Yeh, 2008). This section reviews various methods in detail.

**Iterative estimation**

In iterative estimation, $f_0$ of the predominant source is computed in the first pass, the estimated source and its harmonics are suppressed in the second pass, and the next candidate is pursued. This process is repeated until the termination requirement is met. Iterative estimation assumes that at each iteration, there exists a predominant source such that the extraction of one single $f_0$ is reliable when the remaining partials are fragmentary. Schroeder’s histogram based approach is employed for multipitch estimation
in (Kemp, T. and Waibel, A., 1999). In this work, peak picking in summary autocorrelation function (SACF) is used as $f_0$ candidate in the first pass. In the second pass, it is cancelled from the autocorrelation function (ACF). Iterative cancellation is also applied in the spectral domain for multipitch estimation (Berenguer et al., 2005), in which binary masks around matched harmonics are employed to annihilate the estimated partials from the spectrum. In Klapuri’s work (Klapuri, 2008), signals are first compressed and half wave rectified. Harmonic matching in the summary magnitude spectrum results in the predominant pitch followed by its attenuation. Spectral smoothness principle was proposed as an efficient new mechanism in multipitch estimation in (P.Klapuri, 2001). Spectral smoothness refers to the expectation that the spectral envelopes of real sound sources tend to be continuous. The experimental framework of the spectral-smoothness based model is given in Figure 4.2.

A graphical model based framework is employed in (Jordan, 2004) for pitch estimation. Maximization of likelihood in a spline smoothing model leads to predominant pitch detection in the proposed work. In (Abeysekera, 2004), two dimensional distribution is computed using bispectrum for pitch estimation. The estimated pitch component is annihilated from the mixed source. Ming Li et al. proposed an iterative estimation method grounded on the subharmonic summation method and spectral cancellation framework (Li et al., 2008). Iterative strategy in spectral domain is attempted in (Kemp, T. and Waibel, A., 1999). First, $f_0$ derived from spectral peaks is removed from the spectrum and a second $f_0$ is estimated from the residual. In de Cheveigne’s work, both joint cancellation and iterative cancellations in time domain are studied (de Cheveigne, 1993).

![Figure 4.2: Iterative multipitch estimation (Klapuri, 2008)](image_url)
**Joint estimation**

In joint estimation, multiple fundamental frequencies are jointly estimated without any cancellation framework. A two way mismatch method to estimate pitches is proposed in Maher and Beauchamp (Maher and Beauchamp, 1994). The algorithm searches for the pair of fundamental frequencies that minimizes frequency discrepancies between the harmonic models and the observed peaks. It is the mismatch from the predicted to measured and the mismatch from the measured to the predominant. Each match is weighted by the amplitudes of the observed peaks. In this way the algorithm minimizes the residual by the base match. Some of the algorithms follow a correlogram based approach, in which channel-lag representation of auto-correlation function in the auditory channel is further processed. Enhanced summary autocorrelation function (ESAF) based pitch estimation is used in the proposed work of Karjalainen and Tolonen (Tolonen and Karjalainen, 2000).

The work proposed by Zhang et al. (Zhang et al., 2008) uses weighted summary auto-correlation function (SACF) in which modified amplitude of ACF models the relationship between $f_0$ and frequency of channel. In (de Cheveigne, 1993; Tolonen and Karjalainen, 2000), temporal approaches are used with joint estimation strategy. Meddis and Hewitt extended their single pitch perception model to multiple source separation in (Meddis and Hewitt, 1991). The correlogram based algorithm proposed by Wu et al. (Wu and Wang, 2003) uses a unitary model of pitch perception to estimate the pitch of multiple speakers. The input signal is decomposed into sub-bands using a gammatone filterbank and the framewise normalized autocorrelation function is computed for each channel. The peaks selected from all the channels are used to compute the likelihoods of pitch periodicities and these likelihoods are modeled by an HMM to generate the pitch trajectories.

2-D processing of speech in multipitch estimation is proposed by Tianyu et al. (Wang and Quatieri, 2009). The localized time-frequency regions in the narrow band spectrogram are analyzed to extract pitch candidates in the work. Multi-region analysis combined with the pitch candidates results in the final pitch estimates (Wang and Quatieri, 2009). M.H Radfar et al. proposed a joint estimation method called MP-Tracker in spectral domain (Radfar et al., 2011). Pitches are estimated by minimizing the log spectral distortion between the mixture spectrum and parametric spectra with re-
spect to the underlying frequency components to obtain estimates of the pitch frequencies. Periodicity features are extracted through an auditory front-end and less corrupted channels in (Jin and Wang, 2011). A summary of few algorithms are listed in the Table 4.1.

Table 4.1: Multipitch estimation-Summarization of few algorithms

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Year</th>
<th>Name of Author</th>
<th>Algorithms used</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1999</td>
<td>P.J. Walmsley et al.</td>
<td>Polyphonic pitch tracking using joint Bayesian estimation</td>
</tr>
<tr>
<td>2</td>
<td>2001</td>
<td>A.P. Klappuri (Klapuri, 2001)</td>
<td>Spectral smoothness principle</td>
</tr>
<tr>
<td>3</td>
<td>2003</td>
<td>A.P. Klappuri (Klapuri, 2003)</td>
<td>Based on harmonicity and spectral smoothness</td>
</tr>
<tr>
<td>4</td>
<td>2008</td>
<td>A. Klapuri (Klapuri, 2008)</td>
<td>Using an auditory model</td>
</tr>
<tr>
<td>5</td>
<td>2009</td>
<td>R.Bedaeu et al. (Badeau et al., 2009)</td>
<td>Expectation maximization algorithm</td>
</tr>
<tr>
<td>6</td>
<td>2010</td>
<td>E. Vincent et al. (Vincent et al., 2010)</td>
<td>Adaptive harmonic spectral decomposition</td>
</tr>
<tr>
<td>7</td>
<td>2011</td>
<td>Q. Huang et al. (Huang and Wang, 2011)</td>
<td>Based on multi-length windows harmonic model</td>
</tr>
<tr>
<td>8</td>
<td>2011</td>
<td>Wohlmayr et al. (Wohlmayr et al., 2011)</td>
<td>Using factorial hidden Markov approach</td>
</tr>
<tr>
<td>9</td>
<td>2012</td>
<td>John Xi Zhang et al. (Zhang et al., 2012)</td>
<td>Subspace analysis along with time space model</td>
</tr>
<tr>
<td>10</td>
<td>2012</td>
<td>Daniele Giacobello et al. (Giacobello et al., 2012)</td>
<td>Combination of sparsity and linear prediction framework</td>
</tr>
<tr>
<td>11</td>
<td>2013</td>
<td>S.I. Adalbjornsson (Adalbjornsson et al., 2013)</td>
<td>Using block sparsity</td>
</tr>
<tr>
<td>12</td>
<td>2014</td>
<td>Zhiyano Duan et al. (Pardo et al., 2010)</td>
<td>Constrained clustering approach</td>
</tr>
</tbody>
</table>

4.4 Evaluation methodology

The guidelines for evaluating the performance of monopitch estimation can be found in (Rabiner et al., 1976). Since there are no guidelines for the performance evaluation in the case of multipitch tracking, usually the guidelines of single pitch tracking is extended for multipitch estimation. Speech mixtures are obtained by mixing utterances at 0 dB. The utterances are chosen from any of the monopitch databases. The ground truth is computed using one of the efficient mono-pitch tracking algorithms available.

Datasets

Datasets, which are used in the multipitch experiments include GRID (Cooke et al., 2006), ATR Database (A.Kurematsu et al., 1995), PTDB-TUG Database (Petriker et al.,
Evaluation metrics

The metrics used for the evaluation of the performance of various multipitch estimation algorithms are listed below.

- **Accuracy**: $Accuracy_{10}$ and $Accuracy_{20}$ correspond to the percentage of frames at which pitch deviation is less than 10% and 20% with respect to the reference, respectively. A gross error occurs if the detected pitch is not within the specified threshold with respect to the reference pitch.

- **Standard deviation of the fine pitch errors ($E_{fs}$)**: The standard deviation of the fine pitch error is a measure of the accuracy of pitch detection during voiced intervals. The standard deviation of the pitch detection ($E_{fs}$) is given by:

  $$E_{fs} = \sqrt{\frac{1}{N} \sum (p_s - p'_s)^2 - e^2}$$  

  where $p_s$ is the standard pitch, $p'_s$ is the detected pitch, $N$ is the number of correct pitch frames and $e$ is the mean of the fine pitch error. $e$ is given by:

  $$e = \frac{1}{N} \sum (p_s - p'_s)$$

- **Transition error**: The algorithms proposed in (Wu and Wang, 2003; Jin and Wang, 2011) use misclassification as a criterion along with gross errors. In the error calculation, $E_{x \rightarrow y}$ denotes the error rate of time frames where $x$ pitch periods are misclassified as $y$ pitch periods. The gross error, $E_{gs}$ is the percentage of frames where the detected pitch differs from the true pitch by more than 20%.

4.5 Applications of multipitch estimation

The estimation of the fundamental frequency, or the pitch of audio signals has a wide range of applications in CASA, prosody analysis, source separation and speaker identification (de Cheveigne, 1993; Murthy and Yegnanarayana, 2011). In music context, multipitch estimation is inevitable in applications such as the extraction of “predominant $f_0$” (Salamon and Gomez, 2012), computation of bass line (Goto and Hayamizu, 1999), content-based indexing of audio databases (Tao Li et al., 2003) and automatic
transcription (Ryynanen and Klapuri, 2008a). It should be noted that interactive music applications demand highly robust real time pitch estimation algorithms in all aspects (Poliner et al., 2007). Multipitch estimation is widely used in source separation (P.Klapuri, 2001; Wang and Loizou, 2006).

In (Wang and Loizou, 2006), a single channel speech separation method based on binary spectrum mask estimation is proposed. The long-short frame associated harmonic model (LSAH) is used to analyze the spectrum for a speech mixture. Then multipitch estimation and binary spectrum mask estimation are performed for each speech source. In (Shao and Wang, 2003), a new usable speech extraction method is proposed to improve speaker identification performance under the co-channel situation based on pitch information obtained from a robust multipitch tracking algorithm. Source separation through multipitch estimation is addressed by spectral smoothness principle in (P.Klapuri, 2001).

Multipitch estimation has also established its role in music information retrieval applications. In melody extraction task proposed by Salamon et al. (Salamon and Gomez, 2012), several pitch contours are first created using salience function and then melody is estimated based on contour characteristics. In (Salamon et al., 2012a), a method is proposed for tonic identification for Indian classical music based on a multipitch analysis of the audio signal. Unlike approaches that identify the tonic from a single predominant pitch track, multipitch representation is used to construct a pitch histogram of the audio excerpt from which the tonic is identified (Salamon et al., 2012a). Multipitch estimation task has also become a central tool in musical scene analysis (Christensen and Jakobsson, 2009).

### 4.6 Modified group delay based two-pitch tracking in co-channel speech

Majority of the pitch tracking methods are usually limited to clean speech and give a degraded performance in the presence of other speakers or noise. An algorithm based on modified group delay functions is proposed and evaluated on various datasets in this work.
4.6.1 Theory of pitch detection using modified group delay functions

The monopitch estimation discussed in Section 3.9 can be extended to estimate multiple pitches in multi-talker environment. In the case of multiple speakers, the flattened power spectrum contains the excitation information of all the speakers.

Consider a speech signal with two different pitches corresponding to that of two different voices. The source spectrum of this signal (assuming no linear phase) is

\[ E(z) = 1 + z^{-T_o} + z^{-T_1} + z^{-2T_o} + z^{-2T_1} + z^{-3T_o} + z^{-3T_1} + \ldots \]  

(4.4)

Assuming that a frame of speech contains no more than two pitch periods, and that there is no linear phase, the power spectrum of the source is given by:

\[ E(z)E^*(z) = (1 + z^{-T_o} + z^{-T_1} + z^{-2T_o} + z^{-2T_1})(1 + z^{T_o} + z^{T_1} + z^{2T_o} + z^{2T_1}) \]  

(4.5)

Substituting \( z = e^{j\omega} \),

\[ |E(e^{j\omega})|^2 = 5 + 4 \cos(\omega T_o) + 4 \cos(\omega T_1) + \\
2 \cos(\omega 2T_o) + 2 \cos(\omega 2T_1) + 2 \cos(\omega(T_o - 2T_1)) + \\
2 \cos(\omega(T_1 - 2T_o)) + 2 \cos(\omega(2(T_1 - T_o))) + 2 \cos(\omega(T_1 - T_o)) \]  

(4.6)

By introducing a parameter \( \gamma \quad (0 < \gamma \leq 1) \), to control the flatness of the spectrum, we have

\[ |E(e^{j\omega})|^2\gamma = (5 + 4 \cos(\omega T_o) + 4 \cos(\omega T_1) + \\
2 \cos(\omega 2T_o) + 2 \cos(\omega 2T_1) + 2 \cos(\omega(T_o - 2T_1)) + \\
2 \cos(\omega(T_1 - 2T_o)) + 2 \cos(\omega(2(T_1 - T_o))) + 2 \cos(\omega(T_1 - T_o)))^\gamma \]  

(4.7)

This only results in a composite signal which still is a sum of sinusoids. The number of
sinusoids and the frequencies of the sinusoids are related to the pitch of the individual
sources. If the spectral components corresponding to the periodic component are em-
phasised, the problem of pitch extraction reduces to that of the estimation of sinusoids
in the frequency domain. We now replace $\omega$ by $n$ and $T_o, T_1$ by $\omega_o, \omega_1$, respectively in
Equation (4.7), and remove the dc component to obtain a signal which is ideally a sum
of sinusoids that corresponds to the excitation components of the mixture.

$$s[n] = a \cos(n\omega_o) + b \cos(n\omega_1) + c \cos(n2\omega_o)$$
$$+ d \cos(n2\omega_1) + e \cos(n(\omega_0 - 2\omega_1)) + f \cos(n(\omega_1 - 2\omega_0))$$
$$+ g \cos(n2(\omega_1 - \omega_0)) + h \cos(n(\omega_1 - \omega_0)) + \cdots + \cdots$$

where $a, b, c, d, e, f, g, h$ are constants.

This signal is subjected to modified group delay processing, which results in peaks
at multiples of the various pitches present in the speech mixture. The procedure to map
these peak locations to constituent pitch trajectories is explained in the next section.

### 4.6.2 Proposed system description

The block diagram of the proposed system is given in Figure 4.3. The dotted block
represents the MODGD (Source) based predominant pitch estimation algorithm dis-
cussed in Chapter 3. The proposed algorithm computes constituent pitch frequencies
from the mixed speech through iterative estimation and cancellation method. First, the
flattened spectrum is frame-wise analysed by MODGD algorithm. As discussed in Sec-
tion 3.9, peaks can be observed in the MODGD domain at locations corresponding to
the multiples of all the pitch periods and its algebraic combinations. As discussed in the
prominent pitch estimation algorithm, the location of the prominent peak in the range
$[P_{\text{min}}, P_{\text{max}}]$ in the MODGD feature space is mapped to the first candidate pitch esti-
mate, where $P_{\text{min}}, P_{\text{max}}$ represent minimum and maximum pitch, respectively. In the
second pass, the estimated pitch component and its harmonics are annihilated from the
flattened power spectrum. The residual signal is again subjected to MODGD analysis. In
the post processing phase, pitch grouping followed by removal of pitch outliers results
in the individual pitch trajectories.
Figure 4.3: Block diagram of the proposed method. Dotted block represents the MODGD (Source) based predominant pitch estimation algorithm.

4.6.3 Estimation of constituent pitches

Once the prominent pitch is estimated from the flattened spectrum in the first pass, the objective is to estimate the second pitch candidate. The proposed algorithm uses comb filters to annihilate the estimated pitch and its harmonics from the flattened spectrum in the second pass.

Comb filtering

Comb filters are widely used in many speech processing applications such as speech enhancement, pitch detection and speaker recognition (Jin et al., 2010; Laskowski and Jinn, 2010). It has already been shown that the optimum-comb technique is a fast and useful technique for the extraction of pitch period data from continuous speech in (Moorer, 1974). Adaptive comb filter (ACF) for harmonic signal enhancement and spectral estimation is proposed in (Porat and Nehorai, 1986). In (Tan and Alwan, 2011), a signal to noise ratio weighted correlogram based pitch estimation algorithm is described. A bank of comb filters operates in each of the low, mid and high frequency bands to capture different set of harmonics. Comb filters are also utilized for musical pitch estimation and the discrimination of different musical instruments (Miwa et al., 2000). The proposed algorithm uses comb filters to annihilate the estimated pitch and
its harmonics from the flattened spectrum in the first pass. If a signal is periodic, it suggests that some basic waveform repeats with a certain frequency. Mathematically, we can express it as \(x[n] \approx x[n - D]\) where \(D\) is the repetition or pitch period. A measure of periodicity can be obtained using a metric \(e(n)\) defined as

\[
e[n] = x[n] - \alpha x[n - D]
\] (4.9)

where \(\alpha\) is a constant, introduced to allow for some variation of the amplitude.

In \(Z\) domain,

\[
E(z) = (1 - \alpha z^{-D})X(z)
\] (4.10)

The FIR comb filter transfer function is represented as:

\[
H(z) = \frac{E(z)}{X(z)} = 1 - \alpha z^{-D}
\] (4.11)

which shows that the process of matching of a signal by a delayed version of itself is a filtering problem. Notching comb filters are filters that cancel out signal components at certain frequencies. In the complex case, these typically have the following form

\[
H(z) = \frac{1 + \beta z^{-1}}{1 + \rho \beta z^{-1}} = \frac{P(z)}{P(\rho^{-1} z)}
\] (4.12)

where \(\beta\) is a complex coefficient and \(-1 < \rho < 1\) is real. Since the periodic signal is comprised of possibly many harmonics, we can use \(L_k\) such notch filters having notches at frequencies \(\Psi_i\). Such a filter can be factorized into the following form.

\[
P(z) = \prod_{i=1}^{L_k} (1 - e^{i\Psi_i} z^{-1})
\] (4.13)

\[
P(z) = 1 + \beta_1 z^{-1} + \beta_2 z^{-2} + \cdots + \beta_{Lk} z^{-Lk}
\] (4.14)

which has zeros on the unit circles at angles corresponding to the desired frequencies.

The comb effect results from phase cancellation and reinforcement between the delayed and undelayed signal. The magnitude response of the comb filter is

\[
|H(e^{j\omega})| = \sqrt{(1 + \alpha^2) + 2\alpha \cos(\omega D)}
\] (4.15)

Magnitude response and pole-zero plot of comb filter is shown in Figure 4.4. In the
proposed approach, comb filter is used to annihilate the predominant fundamental frequency component obtained in the first pass from a composite flattened power spectrum which constitutes multiple excitations.

**Removal of partials**

Consider a speech mixture of two synthetic speech signals with fundamental frequencies of 200 Hz and 280 Hz. One frame of such a mixed speech is shown in Figure 4.5 (a). Flattened spectrum of corresponding frame is shown in Figure 4.5 (b). Modified group delay function computed for the synthetic mixture frame is shown in Figure 4.5 (c). The peaks in MODGD feature space, correspond to the pitch candidates present in the speech mixture and its integral multiples. In the first pass, the prominent peak in the MODGD feature space is mapped to first pitch estimate followed by the annihilation of it from the residual spectrum. The red color contour in Figure 4.5 (d) is the computed MODGD for the second pass. The individual pitch tracks computed through the steps discussed are shown in Figure 4.5 (e) along with reference pitches. Similarly, another natural audio mixture example is shown in Figure 4.6. Figure 4.6 (a) shows the MODGD plot for a real audio frame and in Figure 4.6 (b), pitch estimates of the audio segment are shown. The modified group delay functions obtained in the first pass, and in the second pass are illustrated in the figure. It is obvious from the figure that the pitch
component corresponding to the predominant peak is annihilated in the second pass.

![Graphs](image)

Figure 4.5: (a) A frame of synthetic mixture, (b) Flattened spectrum of frame in (a), (c) MODGD plot in the first pass, (d) MODGD of (a) after the application of comb filter (second pass), (e) Pitch extracted for the mixed synthetic speech segment.

### 4.6.4 Pitch trajectory estimation by grouping

At the end of the pitch estimation phase, two pitch candidates per frame are computed. In the pitch grouping stage, these candidates are grouped into trajectories which comprise of continuous smooth individual tracks. Since pitch crossing is not considered, out of two candidates per frame, high pitch values are consolidated to form one trajectory, while smaller values to the other. Alternatively, dynamic programming based pitch grouping can also be employed. In that case, the relative closeness of the distance be-
between peaks in two consecutive frames is used to compute the optimal path. Transition cost is computed as the absolute difference in distance between the current and previous frame. The optimal path is selected by minimizing the transition cost across frames using backtracking. The transition cost $C_t(c_j/c_{j-1})$ between the pitch candidates $c_j$ and $c_{j-1}$ of consecutive frames is given by (Veldhuis, 2000)

$$C_t(c_j/c_{j-1}) = | L_j - L_{j-1} | \quad (4.16)$$

where $L_j$, $L_{j-1}$ are peak locations in consecutive frames. The optimal pitch sequence $(c_1...c_M)$ is computed using dynamic programming with candidates $c_1$ in the first and $c_M$ in the $M^{th}$ frame in a block by minimizing the transition cost function (Veldhuis, 2000). Transition cost $TC(c_1...c_M)$ of pitch candidates $c_1$ to $c_M$ is computed by

$$TC(c_1...c_M) = \sum_{j=2}^{j=M} C_t(c_j/c_{j-1}) \quad (4.17)$$
The optimal sequence of pitch markers is determined by back tracking from the candidate $c_M$ in the $M^{th}$ frame in a block to $c_1$ in the first frame. If the pitch detection algorithm computes any spurious candidate, dynamic programming may result in erroneous pitch tracks. Continuity of pitch contours is used as a criteria to disambiguate the pitch values.

![Figure 4.7](image.png)

**Figure 4.7:** (a) Harmonic energy of a segment, (b) Initial pitch estimates of the segment in (a), (c) Final pitch trajectory in the post processing stage.

### 4.6.5 Postprocessing

The accuracy in pitch estimation is improved by a post processing stage in which the first task is to identify the segments where one speaker is alone present. A soft threshold on harmonic energy is employed to identify these segments. Harmonic energy is computed from the estimated fundamental frequency and its harmonics. Multiples of
fundamental frequency are obtained by searching for local maxima with 3% tolerance. Harmonic energy of a signal $x[n]$ is computed by (Ramakrishnan et al., 2008)

$$E_n = \sum_{k=k_f_0}^{k_{N_f_0}} |X[k]|^2,$$

where $X[k]$, $k$, $f_0$ represent the Fourier transform magnitude, bin number, fundamental frequency, respectively. $k_{f_0}$ and $k_{N_f_0}$ represent bins corresponding to fundamental frequency and its $N^{th}$ multiple. Segments which are detected as single speaker frames in the voicing detection stage are processed again for monopitch estimation using the MODGD algorithm. If the pitch estimated follows a path, the estimated sequence is appended to the existing pitch contour by ensuring consistency. This is illustrated in Figure 4.7. Figure 4.7 (a) shows harmonic energy plotted for a mixed speech segment. Regions 1 and 3 in Figure 4.7 (b) consists of only one speaker speech, these regions have be processed using monopitch estimation algorithm. The appended pitch tracks obtained are shown in Figure 4.7 (c).

As part of smoothening the curve, stray values are removed by framing appropriate rules that refine pitch estimates. For example, let $f_t$ and $f_{t+1}$ be the pitch candidates of consecutive frames in a pitch track after the grouping stage. If $f_{t+1}$ lies outside the range $[f_t - \rho, f_t + \rho]$, this is treated as a spurious pitch estimate and will be interpolated using previous and successive pitch values (Radfar et al., 2011). We use linear interpolation for replacing missing pitch frequencies; however, other interpolation techniques such as cubic or spline interpolation could be used. This simple but effective technique reduces the pitch error considerably. Note that missing pitch frequencies should typically not be interpolated for segments corresponding to 40 ms (Radfar et al., 2011). The threshold $\rho$ is set heuristically to 10 Hz. A typical example is shown in Figure 4.6(b). The circled part indicates the presence of two stray values in the middle of a continuous curve. The estimated pitch trajectories for a speech mixture consisting of male and female utterances are shown in Figure 4.8. Figure 4.8 (a) shows the initial pitch estimates and Figure 4.8 (b) shows the individual pitch trajectories after post processing. The pitch trajectories estimated using Wu et al. algorithm (D.L.Wang et al., 2003) are also shown in Figure 4.8 (c) for the same speech mixture.
Figure 4.8: (a) Initial pitch estimates for a speech mixture, (b) Final pitch trajectories estimated using the proposed algorithm, (c) Pitch trajectories estimated using WWB algorithm.

4.6.6 Pitch extraction in noisy and reverberant environment

The presence of noise and reverberation in speech poses a problem, even for monopitch estimation. For noise corrupted speech, both the time-domain periodicity and spectral-domain periodicity are distorted and hence, conventional pitch estimation fails to a certain extent (Huang and Lee, 2013). Group delay domain representation of speech makes it relatively immune to noise when compared to that of the short-time magnitude spectrum (Hegde et al., 2007b; Yegnanarayana and Murthy, 1992). For instance, consider the noisy signal $x[n]$ as the output of the autoregressive process $s[n]$, corrupted with Gaussian noise $\omega[n]$, i.e

$$x[n] = s[n] + \omega[n]$$  \hspace{1cm} (4.19)

Group delay analysis of an autoregressive process in a noisy environment is discussed in (Yegnanarayana and Murthy, 1992). $z$-transform of $s[n]$, ignoring the effects of
truncation of the response of an all-pole system is given as

\[ S(z) = \frac{GE(z)}{A(z)} \]  \hspace{1cm} (4.20)

where \( E(z) \) is the z-transform of the excitation sequence \( e[n] \) and \( G/A(z) \) is the z transform of the all-pole system corresponding to the autoregressive process. From Equation 4.19 and Equation 4.20, we get;

\[ X(z) = \frac{GE(z) + W(z)A(z)}{A(z)} = \frac{V(z)}{A(z)} \]  \hspace{1cm} (4.21)

In group delay domain,

\[ \tau_X(e^{j\omega}) = \tau_V(e^{j\omega}) - \tau_A(e^{j\omega}) \]  \hspace{1cm} (4.22)

As explained in (Yegnanarayana and Murthy, 1992), the noise spikes in \( \tau_X(e^{j\omega}) \) can be suppressed by multiplying with the estimated zero spectrum. This results in an estimate of \( -\tau_A(e^{j\omega}) \) which corresponds to the spectral component in the composite signal. Thus, group delay based approach is very effective in analyzing frequency components of a composite signal in the presence of noise. Room reverberation adversely affects the characteristics of pitch and thus makes the task of pitch determination more challenging. It causes degradation of the excitation signal (Jin and Wang, 2010). In reverberant environments, the speech signal that reaches the microphone is superimposed with multiple reflected versions of the original speech signal. These superpositions can be modeled by the convolution of the room impulse response (RIR), that accounts for individual reflection delays, with the original speech signal (Allen and Berkley, 1979). Mathematically, the reverberant speech \( r[n] \) is obtained as the convolution of speech signal \( s[n] \) and room impulse response \( h[n] \) (Thomas et al., 2008).

\[ r[n] = s[n] * h[n] \]  \hspace{1cm} (4.23)

The room impulse response is described as one realization of a non-stationary stochastic process in Schroeder’s frequency-domain model (Jot et al., 1997) given by,

\[ h[n] = b[n]e^{-\delta n} \]  \hspace{1cm} (4.24)

where \( b[n] \) is a centered stationary Gaussian noise, and \( \delta \) is related to the reverberation
time $T_r$. A typical room impulse response used for the experiment is shown in Figure 4.9. The proposed algorithm is also analysed in a reverberant environment using simulated impulse response.

![Figure 4.9: Room impulse response](image)

**Table 4.2: Category of mixtures for Dataset:1 and 2**

<table>
<thead>
<tr>
<th>Category</th>
<th>Speech data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Male/Female, Female/Female, Male/Male</td>
</tr>
<tr>
<td>2</td>
<td>Male/Female, babble noise</td>
</tr>
<tr>
<td>3</td>
<td>Male/Female, white noise</td>
</tr>
<tr>
<td>4</td>
<td>Male/Female with reverberation</td>
</tr>
</tbody>
</table>

### 4.7 Performance evaluation

In the proposed work, focus is given to the multipitch estimation of speech mixture with two speakers. The performance of the proposed algorithm was evaluated using the following datasets:

- **Dataset-1**: The dataset consists of 40 audio files obtained by mixing a subset of utterances from the Pitch Tracking Database of Graz university of Technology (PTDB-TUG) (Petriker et al., 2011), Childers database and a few audio samples from *Simple* $^4$ *All* speech corpora (Suni et al., 2014). The PTDB-TUG consists of audio recordings with phonetically rich sentences from TIMIT corpus. *Simple* $^4$ *All* corpora consists of audio samples from different languages.

- **Dataset-2**: GRID (Cooke et al., 2006) corpus consists of high-quality audio and video recordings of 1000 sentences spoken by each of 34 talkers (18 male, 16
female). A subset of 40 audio files are used for generating mixtures for the evaluation.

In the experiments, each audio mixture is processed using a Hamming window of frame length of 30 ms and hop size of 10 ms. As shown in Table 4.2, the interferences are classified into four categories by considering clean and noise conditions. The data set contains audio files of cross gender (male/female) and same gender (female/female, male/male). The test was conducted mainly for a 0 dB target-to-masker ratio (TMR) which is considered as the most difficult situation in co-channel speech segregation problem as both talkers equally mask each other. In categories 2 and 3, speech is obtained by mixing the category 1 speech data with babble noise (5 dB SNR) and white noise (10 dB SNR). Category 4 interferences comprising of simulated reverberant speech utterances. The performance is also evaluated with speech mixture of male and female voices with +3 dB and -3 dB TMR. Reverberant speech is generated using simulated room acoustics using the model proposed in (Allen and Berkley, 1979). The simulation uses a reverberation time of $T_{60} = 200$ ms.

Table 4.3: Performance evaluation on Dataset-1

<table>
<thead>
<tr>
<th>Category</th>
<th>MODGD</th>
<th>WWB</th>
<th>JIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male-Female</td>
<td>84.58</td>
<td>76.71</td>
<td>81.04</td>
</tr>
<tr>
<td>Female-Female</td>
<td>75.02</td>
<td>60.78</td>
<td>70.31</td>
</tr>
<tr>
<td>Male-Male</td>
<td>73.58</td>
<td>66.01</td>
<td>70.01</td>
</tr>
<tr>
<td>Male-Female,Babble noise (10 dB SNR)</td>
<td>72.12</td>
<td>60.17</td>
<td>73.43</td>
</tr>
<tr>
<td>Male-Female,Babble noise (5 dB SNR)</td>
<td>70.12</td>
<td>52.00</td>
<td>66.72</td>
</tr>
<tr>
<td>Male-Female,White noise (10 dB SNR)</td>
<td>73.13</td>
<td>62.97</td>
<td>76.21</td>
</tr>
<tr>
<td>Male-Female,White noise (5 dB SNR)</td>
<td>68.05</td>
<td>55.24</td>
<td>66.50</td>
</tr>
<tr>
<td>Male-Female with reverberation</td>
<td>63.05</td>
<td>62.70</td>
<td>65.28</td>
</tr>
</tbody>
</table>

Table 4.4: Performance evaluation on Dataset-2

<table>
<thead>
<tr>
<th>Category</th>
<th>MODGD</th>
<th>WWB</th>
<th>JIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male-Female</td>
<td>82.65</td>
<td>78.92</td>
<td>78.95</td>
</tr>
<tr>
<td>Female-Female</td>
<td>75.01</td>
<td>76.74</td>
<td>76.81</td>
</tr>
<tr>
<td>Male-Male</td>
<td>70.30</td>
<td>50.84</td>
<td>73.08</td>
</tr>
<tr>
<td>Male-Female,Babble noise (10 dB SNR)</td>
<td>74.32</td>
<td>56.25</td>
<td>72.74</td>
</tr>
<tr>
<td>Male-Female,Babble noise (5 dB SNR)</td>
<td>71.08</td>
<td>50.30</td>
<td>65.08</td>
</tr>
<tr>
<td>Male-Female,White noise (10 dB SNR)</td>
<td>66.50</td>
<td>66.34</td>
<td>71.64</td>
</tr>
<tr>
<td>Male-Female,White noise (5 dB SNR)</td>
<td>59.20</td>
<td>55.28</td>
<td>60.64</td>
</tr>
<tr>
<td>Male-Female with reverberation</td>
<td>68.00</td>
<td>64.01</td>
<td>70.38</td>
</tr>
</tbody>
</table>

The performance is evaluated only for voiced frames. The reference frequency of an unvoiced frame is considered as 0 Hz. To evaluate the performance of our algorithm, a reference pitch contour corresponding to true individual pitch is required. We
computed the reference pitch of clean speech using Wavesurfer (W.S-URL, 2012). The performance is quantitatively assessed by measuring two types of metrics: accuracy and standard deviation of the fine pitch errors $E_{fs}$.

### 4.8 Results analysis and discussions

The performance of the proposed algorithm is evaluated on three different speech mixtures, namely, same gender (M/M, F/F) and cross gender (M/F). In addition to the clean speech condition, the performance is also evaluated in noisy and reverberant conditions. WWB algorithm (D.L.Wang et al., 2003) and Jin et al. algorithm (Jin and Wang, 2011) are used for objective comparison. The algorithm of Wu, Wang, and Brown is referred to as the WWB algorithm. WWB algorithm integrates a channel-peak selection method and HMM for forming continuous pitch tracks. WWB framework computes final pitch estimates in three stages: auditory front-end-processing, pitch statistical modelling and HMM tracking. Jin et al. algorithm, designed specially to tackle reverberant noises is similar to WWB algorithm but different in channel selection and pitch scoring strategy. In (D.L.Wang et al., 2003; Jin and Wang, 2011), half of the corpus is used to estimate the model parameters for further analysis. Another important fact about the experiments reported in (D.L.Wang et al., 2003) is that they are focused on speech mixtures with one dominant speaker. Both algorithms report considerable transition errors, in which pitch estimates of speaker-1 are misclassified as the pitch estimates of speaker-2.

For a fair comparison, the WWB and Jin et al. algorithm outputs are grouped in the post processing stage to ensure no transition errors occur (Jin and Wang, 2011). The grouping is performed using an approach similar to the approach proposed in (Rafar et al., 2011). The results obtained through the quantitative evaluation are listed in Tables 4.3 - 4.8. Tables 4.3 and 4.4 compare the pitch accuracies with 10% tolerance for dataset-1 and dataset-2, respectively. Jin et al. algorithm and MODGD algorithm show similar performance with MODGD having a slight edge. It was also observed that WWB algorithm failed to pick one of the pitch estimates in many frames. The proposed method reports accuracies of 84.58% and 82.65% for dataset-1 and dataset-2 respectively in clean cross-gender mixtures. It is worth noting that, in babble and white noise conditions, the performance of MODGD system is at par with that of Jin et al. algorithm while it is superior in performance when compared with that of WWB algo-
algorithm. From Tables 4.3 and 4.4, it is seen that as SNR decreases from 10 dB to 5 dB, the proposed algorithm is more robust than WWB and Jin et al. algorithms.

When the gender mixture is made up of the same sex, if the pitch values are too close, the performance of the proposed algorithm is affected due to the filtering operation. Figure 4.10 shows a situation where the speech mixture comprises of speakers with close individual pitches. During the cancellation process, the second pitch is also eliminated and results in a spurious pitch estimate during the next iteration. In the proposed group delay based system, both noise and source introduce zeroes that are close to the unit circle in the $z$ domain (Murthy, 1991; Hegde, Rajesh M., 2005). The fundamental difference is that source zeroes are periodic while noise zeroes are aperiodic. This is the primary reason for the better performance of group delay in noisy environments as compared to other model based methods. Tables 4.5 and 4.6 compare the standard deviation of fine pitch error ($E_{fs}$) for dataset-1 and dataset-2, respectively. While WWB and Jin et al. algorithms report $E_{fs}$ in the range 2 - 4.5 Hz across categories, the proposed algorithm reports higher error. This is primarily due to the resolution that results from the length of the flattened signal. The performance of the proposed system with varying TMR is shown in Table 4.7 and Table 4.8. The analysis shows a similar trend in the performance of both MODGD algorithm and Jin et al. algorithm. A considerable
variation in accuracy is reported in the case of WWB algorithm as TMR decreases from -3dB to 3dB, but Jin et al. and the proposed MODGD algorithm show consistently good performance.

Table 4.7: Comparison of accuracy for different TMR : Database:1

<table>
<thead>
<tr>
<th>Category</th>
<th>MODGD</th>
<th>WWB</th>
<th>JIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (in %)</td>
<td>$E_{t_s}$</td>
<td>Accuracy (in %)</td>
<td>$E_{t_s}$</td>
</tr>
<tr>
<td>Male-Female 0dB</td>
<td>84.58 4.80</td>
<td>76.71 1.81</td>
<td>81.04 3.12</td>
</tr>
<tr>
<td>Male-Female -3dB</td>
<td>83.32 4.93</td>
<td>72.90 1.80</td>
<td>79.65 3.74</td>
</tr>
<tr>
<td>Male-Female +3dB</td>
<td>85.52 4.85</td>
<td>79.40 1.71</td>
<td>82.21 3.42</td>
</tr>
</tbody>
</table>

Table 4.8: Comparison of accuracy for different TMR : Database:2

<table>
<thead>
<tr>
<th>Category</th>
<th>MODGD</th>
<th>WWB</th>
<th>JIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (in %)</td>
<td>$E_{t_s}$</td>
<td>Accuracy (in %)</td>
<td>$E_{t_s}$</td>
</tr>
<tr>
<td>Male-Female 0dB</td>
<td>82.65 3.48</td>
<td>78.92 1.59</td>
<td>78.95 2.37</td>
</tr>
<tr>
<td>Male-Female -3dB</td>
<td>82.72 3.46</td>
<td>71.02 1.73</td>
<td>76.71 2.05</td>
</tr>
<tr>
<td>Male-Female +3dB</td>
<td>82.26 3.49</td>
<td>74.37 1.61</td>
<td>77.61 2.10</td>
</tr>
</tbody>
</table>

Figure 4.10: Speakers with two closed pitch trajectories

4.9 Pitch estimation : more than two speakers

The iterative method can be extended for pitch estimation of mixed speech with more than two speakers. This is illustrated with an example of mixed speech with three speakers. Consider a speech mixture of three synthetic speech signals with fundamental frequencies of 100 Hz, 200 Hz and 280 Hz. Modified group delay function computed for a frame of speech mixture is shown in Figure 4.11 (a). The peaks in MODGD feature space, correspond to the pitch candidates present in the speech mixture and its integral multiples. In the first pass, the prominent peak in the MODGD feature space
is mapped to first pitch estimate followed by the annihilation of it from the residual spectrum. The red color contour in Figure 4.11 (a) is the computed MODGD for the second pass. The green color contour in Figure 4.11 (a) is the computed MODGD for the third pass. The individual pitch estimates computed, in all the iterations are plotted in Figure 4.11 (b) along with reference pitches. The individual pitches estimated using the proposed method for a three speaker natural mixed speech is shown in Figure 4.12.

Figure 4.11: (a) Modified group delay function computed for a speech frame, (b) Pitch contours computed for the speech segment.

Figure 4.12: Pitch contours computed for a natural speech mixture with three speakers.
Table 4.9: Pitch estimation accuracy for three speaker mixed speech.

<table>
<thead>
<tr>
<th>Sl.No</th>
<th>Speech Mixture</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Two females, One male</td>
<td>62.72</td>
</tr>
<tr>
<td>2</td>
<td>Two males, One female</td>
<td>58.72</td>
</tr>
<tr>
<td>3</td>
<td>Three females</td>
<td>50.47</td>
</tr>
<tr>
<td>4</td>
<td>Three males</td>
<td>46.53</td>
</tr>
</tbody>
</table>

The performance of the proposed group delay based iterative method was evaluated using natural three-speaker mixed speech created using GRID subset. Four categories of speech mixtures were considered for the experiment, which include (2 females, 1 male), (2 males, 1 female), (3 females) and (3 males) speech mixtures. The results are tabulated in Table 4.9. The overall accuracy of 54.61% is reported for the entire subset. It is worth noting that, in three speaker case, the experiment is conducted only for pitch estimation, not for tracking. Speaker specific characteristics are needed for accurate tracking of individual pitch contours. Due to iterative cancellation using comb filters, if one of the frequencies in the mixture is a multiple of the other, it may be attenuated on the fly.

**Summary**

Phase based approach for two-pitch tracking is presented, and it yields competitive performance as compared to other state-of-the-art approaches. The flattened spectrum is processed using MODGD algorithm to estimate the predominant pitch in each frame in the first pass. Then the estimated pitch and its harmonics are filtered out using comb filter. In the second pass, the residual spectrum is again analysed using MODGD algorithm to estimate the second candidate pitch. The pitch grouping stage followed by the post processing step result in the final pitch trajectories. The performance of the proposed algorithm was evaluated on mixed speech with cross gender (Female, Male), same gender (Male/Male, Female/Female) mixtures. It does not require pre-training on source models from isolated recordings. The experiment can be extended for multipitch estimation, but for pitch tracking, additional information such as speaker-specific characteristics are needed.
CHAPTER 5

The Source-MODGD Coefficient Features (SMCF) in Speech/Music Analysis

5.1 Introduction

Over the years, researchers in acoustics have employed many feature extraction techniques to parametrize the speech signal in the front-end of numerous speech based applications. Parametric representation has an important role in speech processing. The objective of feature extraction is two-fold: a) compression, and b) discrimination. It should also be invariant to irrelevant factors such as speaker variations, accent differences, speaking rates and background noises (Xiong, 2009). Most of the time, the feature extraction methods are application specific, features performing well in some applications, may not perform well in others; for example the feature extraction methods used for speech or speaker recognition are quite different to those used in MIR applications. Feature extraction methods in time, frequency and cepstral domain have been successfully utilized in various speech based applications. As discussed earlier, group delay based features have also been extensively used in speech and music processing applications. In this chapter, a variant of group delay feature is proposed and the effectiveness of the proposed feature is evaluated for speech-music detection and estimating number of speakers in mixed speech. Section 5.2 introduces a new feature derived from the flattened spectrum and lists the steps to compute the same. The task of estimating number of speakers in a mixed speech using new feature is explained in Section 5.3, along with dataset and performance evaluation. Speech-music classification and detection using the proposed feature is described in Section 5.4.

5.2 Source-MODGD coefficient features (SMCF)

The success of MFCC, combined with its robust and cost-effective computation, turned it into a standard choice in speech recognition applications. The cepstral representa-
tion of the speech spectrum provides a good representation of the spectral properties of the signal. Cepstral features are widely used in speech recognition, speech analysis and speech enhancement applications. Linear prediction cepstral coefficients (LPCCs), Bark frequency cepstral coefficient (BFCC), auditory periphery model based gamma-tone frequency cepstral coefficient (GFCC), have been utilized in many speech processing applications.

Earlier, the MODGDF has been effectively employed in speech recognition, speaker recognition (Hegde, Rajesh M., 2005), language recognition and emotion recognition (Dey et al., 2011). Apart from the MODGDF feature (Hegde, Rajesh M., 2005), the proposed feature is derived from the flattened power spectrum. Group delay analysis on the flattened power spectrum leads to excitation components by annihilating the system characteristics.

Since the flattened spectrum consists of source information of multiple speakers, peaks at multiples of pitch periods and its algebraic combinations can be seen in MODGD feature space. Since the musical frame consists of many pitch components due to accompanying instruments other than the singer, the MODGD feature space will have peaks at multiples of each of the pitch periods. This can be observed in Figure 5.1. Figure 5.1 (a) shows MODGD on flattened power spectrum for a music frame and Figure 5.1 (b) shows MODGD on flattened power spectrum for a speech frame. This characteristics can be used as a criteria to distinguish music from speech. The modified group delay functions derived from the flattened spectrum is converted to useful features using DCT. The new feature representation, source-modified group delay coefficient features (SMCF), $c[n]$ are obtained by (Murthy and Rao, 2003),

$$c[n] = \sum_{k=0}^{k=N_f} \tau_{sm}[k] \cos\left(\frac{n(2k+1)\pi}{N_f}\right)$$

where $N_f$ is the discrete Fourier Transform order and $\tau_{sm}[k]$ is the source modified group delay.

Steps to compute Source-MODGD Coefficient features (SMCF) are summarized below.

- Frame blocking the speech/music signal at a frame size of 20 ms and frame shift of 10 ms.
• Speech/music spectrum is flattened using root cesptral smoothing to annihilate the system characteristics.

• Apply MODGD algorithm on the flattened power spectrum to compute modified group delay function.

• Compute DCT on MODGD spectra to get the SMCF vectors.

5.3 Estimating number of speakers in multipitch environment

Meetings with a number of individuals, can lead to strong interference from several speakers. The information about the number of speakers in such a situation is useful in blind source separation (Emiya et al., 2008), and pitch trajectory estimation in multipitch environment (Wu and Wang, 2003). Estimating the number of speakers plays a crucial role in speaker diarization.

5.3.1 Related work

During radio talk shows, at some point it happens that all the guests speak simultaneously with varying intonation. The ability to track the locations of intermittently speaking multiple speakers in the presence of background noise and reverberation is of great interest due to the vast number of potential applications (Quinlan et al., 2009). Hershey et al, made use of a model-based speaker identification (SID) module, called Iroquois (Hershey et al., 2010) to identify speakers existing in the mixture. In (Shao
et al., 2010), CASA based approach is employed to detect the number of speakers in the speech mixture. MAP criterion is also proposed to detect number of speakers at frame level in model based single channel source separation (SCSS) (Mowlaee et al., 2010). Several sensor based methods were proposed in the literature for detection of the number of sources whose mixed signals are collected from multiple sensors. Multiple sensor based methods rely on the time delays of arrival of speech signals between the two microphones for a given speaker (Swamy et al., 2007; Kumar et al., 2011).

Pitch prediction features (Lewis and Ramachandran, 2001), the temporal evolution of the 2-D Average Magnitude Difference Function (AMDF) (Vishnubhotla and Espy-Wilson, 2008), modulation characteristics (Arai, 2003), the statistical characteristic of the 7th Mel coefficient (Sayoud and Ouamour, 2010) are also utilized in estimating number of speakers. In (Lewis and Ramachandran, 2001), pitch prediction feature (PPF) is used in identifying temporal regions or frames as being either one-speaker or two-speaker speech. A method for multi-speaker tracking using distributed nodes with microphone arrays is described in (Plinge and Fin, 2014). Since the modulation peak decreases as number of speakers increases, algorithm in (Arai, 2003) uses the region of the modulation frequency between 2 and 8 Hz to estimate the number of speakers. In (Emiya et al., 2008), distance between a speech mix and a single speaker reference is used as the metric to compute number of speakers in a speech mix.

5.3.2 Proposed system

In the proposed method, we start from the computation of SMCF from the mixed speech. In the feature extraction phase, 20 dimensional SMCF vectors are frame wise computed. Gaussian mixture model based classifier is used in the classification phase. In the simple Gaussian classifier, each probability density function (PDF) is assumed to be a multidimensional Gaussian distribution whose parameters are estimated using the training set. The iterative expectation-maximization (EM) algorithm is used to estimate the parameters of each Gaussian component and the mixture weights (Munoz-Exposito et al., 2007). In the training phase, feature vectors computed from the training set are used to build models for one-speaker case, two-speaker case and three-speaker case. 64 component GMM is used to model the different classes of speech mixtures. 60% files of the dataset is used for training and the rest for testing. During the testing phase, the clas-
sifier evaluates the log-likelihoods of the unknown speech mixture data against these models. The model that gives the maximum accumulated likelihood is declared as the correct match.

5.3.3 Performance evaluation

The performance of the proposed algorithm is evaluated on speech mixtures generated by the subset of GRID dataset. A subset of 40 audio files are used for generating mixtures for the evaluation. Additive mixtures of the audio files are created using Audacity. The test is conducted on speech mixtures, generated with 0 dB target to interference ratio (TIR). The results are tabulated as a confusion matrix in Table 5.1. The results show that the proposed feature is very effective in discriminating different speakers in a mixed speech. Overall accuracy of the estimation of the number of speakers in the multipitch environment is 74.66%. As the number of speakers increases, more misclassification may be observed. The feature can be made robust using delta and delta-delta coefficients to tackle such situations.

Table 5.1: Confusion matrix for number of speaker estimation task. SP-1, SP-2 and SP-3 denote one speaker case, two speaker case and three speaker case, respectively. Class wise accuracy is given as the last column entry.

<table>
<thead>
<tr>
<th></th>
<th>SP-1</th>
<th>SP-2</th>
<th>SP-3</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SP-1</td>
<td>23</td>
<td>2</td>
<td>1</td>
<td>89</td>
</tr>
<tr>
<td>SP-2</td>
<td>3</td>
<td>15</td>
<td>8</td>
<td>58</td>
</tr>
<tr>
<td>SP-3</td>
<td>1</td>
<td>5</td>
<td>20</td>
<td>77</td>
</tr>
</tbody>
</table>

5.4 Speech/music classification and detection

Another potential application of the proposed feature is in audio indexing. In video and audio databases, archiving is an important task. Presently such archives are indexed manually and can only be accessed through the tedious effort of human experts, whose availability is limited. The important task that is to be carried out is to split the audio file into silence/music/crowd noise segments.

Low bit-rate audio coding can also benefit from speech/music classification. Separate codec designs are used in digital encoding of speech and music. Design of a multi-
mode coder that can accommodate different signals can be efficiently implemented, if speech/music classifier gives appropriate decision on an audio segment (El-Maleh et al., 2000). Segregating the signal into speech and music segments is an obvious first step before applying speech-specific or music-specific algorithm. Speech-Music classification is defined as the binary problem of classifying pre-segmented audio data to the speech or music class. The detection task consists of finding segments of music and speech in a signal and classifying each segment as music or speech. Speech-music detection can be used to segment recordings, which are typically several minutes long in duration.

5.4.1 Related work

Saunders uses statistics of the energy contour and zero-crossing rate to discriminate speech and music (Saunders, 1996). Eric Scheirer and Malcolm Slaney addressed audio segmentation by utilizing various combinations of 13 features (Scheirer and Slaney, 1997). Pfeiffer et al (Pfeiffer and Effelsberg, 1996) use perceptual criteria by matching characteristics such as amplitude, pitch and frequency. A supervised tree-based vector quantizer trained to maximize mutual information (MMI) was attempted in (Foote, 1997). The proposed system in (Samouelian, 1997), uses information theory to construct a decision tree from several different TV programs. During the training phase, the feature extraction framework extracts features from the continuous audio signal on a frame-by-frame basis and uses these features to train a decision tree (Quinlan, 1993). Low frequency modulation (E. Scheirer, 1997), entropy and dynamism features (E. Scheirer, 1997) have also been employed for the audio indexing task.

5.4.2 Speech/music classification

The proposed feature can be effectively utilized for audio indexing in multimedia applications. In the proposed approach, 40 dimensional feature vectors comprising SMCF and delta-SMCF are computed from the audio files and classified using a trained GMM. In Figure 5.2, 20 dimensional SMCF feature vectors for entire frames of speech/music segment are mapped into a two dimensional space. It can be seen that SMCF features are effective in differentiating speech from music.

The GTZAN music/speech data collection (DB1) is used for performance evalua-
tion of the proposed task. The dataset consists of 120 tracks, each 30 seconds long. Each class (music/speech) has 60 examples. The tracks are all 22050 Hz Mono 16-bit audio files in .wav format. The files were collected from a variety of sources including personal CDs, radio, microphone recordings, in order to represent a variety of recording conditions. In speech/music classification task, an audio file is classified into speech/music using GMM framework. 60% of the data is used for training the GMM and the rest for testing during classification stage. The decision to classify an audio file is taken based on the likelihood of entire frames in the utterance. Results are tabulated in Table 5.2. All speech files are identified correctly. The proposed system reports an overall accuracy of 95%.

![Figure 5.2: Two dimensional visualization of speech-music data with SMCF feature using Sammon mapping](image)

Table 5.2: Confusion matrix for audio indexing task. Class wise accuracy is given as the last column entry.

<table>
<thead>
<tr>
<th></th>
<th>Speech</th>
<th>Music</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speech</td>
<td>26</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>Music</td>
<td>2</td>
<td>18</td>
<td>90</td>
</tr>
</tbody>
</table>

### 5.4.3 Speech/music detection

In speech/music detection, the earlier experiment is extended in such a way that system should mark speech/music segments in an audio material. In speech/music detection, experiment is performed on 4 broadcasting materials each of 4 minutes duration (DB2), using a GMM classifier, trained using the SMCF. 30 minutes of speech/music data, col-
Table 5.3: Results on Speech/Music detection on audio recordings. Correctly indexed speech/music frames in each recording is entered in %. Global accuracy is given as the entry in the last column.

<table>
<thead>
<tr>
<th>Audio recording</th>
<th>Speech (%)</th>
<th>Music (%)</th>
<th>Global (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>95.09</td>
<td>74.00</td>
<td>84.54</td>
</tr>
<tr>
<td>2</td>
<td>83.80</td>
<td>90.52</td>
<td>87.16</td>
</tr>
<tr>
<td>3</td>
<td>76.38</td>
<td>88.34</td>
<td>82.36</td>
</tr>
<tr>
<td>4</td>
<td>88.74</td>
<td>78.00</td>
<td>83.37</td>
</tr>
</tbody>
</table>

lected from T.V musical shows are used for training the model. The data is manually annotated.

**Frame level evaluation**

In frame level, the percentage of the audio frames which are correctly recognized is estimated (Theodorou et al., 2012). Frame-based evaluation is carried out on 10 ms segments. It has been observed that silence frames or frames with less pitch content in both the speech and music segments cause overlapping in feature space. To overcome such situations, a frame is classified into speech/music based on the cumulative decision from the neighboring frames up to 0.25 s segment length. To improve the performance accuracy of the speech/music detection, a post-decision processing step is implemented to minimize switching (upto 1 s) between the two classes in a long speech/music segment. Since switching of small segments in between long speech/music is less probable, rules are formulated to avoid such switching in decision making. The preceding and succeeding decisions are used for decision processing of the present segment. Results are tabulated in Table 5.3. From the table, it can be seen that the proposed method could attain global accuracy greater than 80% for all the broadcasting materials.

**Event level evaluation:**

Events\(^1\) will be evaluated on an onset-only basis as well as an onset-offset basis using the precision, recall, and F-Measure. The performance metrics are selected as per the MIREX standards used for speech/music evaluation task in 2015\(^2\). The performance metrics are given by :

---

1. An event is a transition from speech to music or music to speech
• **fb_F**: Frame level F-measure.

• **Onset-only event-based F-measure-500ms tolerance (eb_F_500ms)**: A ground truth segment is assumed to be correctly detected if the system identifies the right class (Music/Speech) and the detected segment’s onset is within a 1000ms range (+/- 500ms) of the onset of the ground truth segment or within 20% of the ground truth segment’s length.

• **Onset-offset event-based F-measure-500ms tolerance (eb_Foff_500ms)**: A ground truth segment is assumed to be correctly detected if the system identifies the right class (Music/Speech), and the detected segment’s onset is within +/- 500ms of the onset of the ground truth segment, and the detected segment’s offset is either within +/- 500ms of the offset of the ground truth segment or within 20% of the ground truth segment’s length.

• **Onset-only event-based F-measure-1s tolerance (eb_F_1s)**: A ground truth segment is assumed to be correctly detected if the system identifies the right class (Music/Speech) and the detected segment’s onset is within (+/- 1000ms) of the onset of the ground truth segment or within 20% of the ground truth segment’s length.

• **Onset-offset event-based F-measure-1s tolerance (eb_Foff_1s)**: A ground truth segment is assumed to be correctly detected if the system identifies the right class (Music/Speech), and the detected segment’s onset is within +/- 1000ms of the onset of the ground truth segment, and the detected segment’s offset is either within +/- 1000ms of the offset of the ground truth segment or within 20% of the ground truth segment’s length.

---

Figure 5.3: Performance evaluation of an audio sample, 1 stands for speech and 2 stands for music
The results of speech/music detection are shown in Table 5.4. The algorithm proposed by Panag et al. (C. Panagiotakis, 2005) and MFCC based system are also shown for comparison. In (C. Panagiotakis, 2005), two signal characteristics are used: the amplitude, measured by the root mean square (RMS), and the mean frequency, measured by the average density of zero-crossings. F-measure of the proposed SMCF based system is higher than MFCC based system but less as compared to the scheme of Panag et al. Figure 5.3, shows an example of speech/music detection in an audio file, using the proposed method.

Table 5.4: Performance comparison of Speech/Music detection in database, DB2

<table>
<thead>
<tr>
<th>Method</th>
<th>fb_F</th>
<th>eb_F_500ms</th>
<th>eb_Foff_500ms</th>
<th>eb_F_1s</th>
<th>eb_Foff_1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Panag et al.</td>
<td>0.8385</td>
<td>0.7040</td>
<td>0.6640</td>
<td>0.7040</td>
<td>0.6640</td>
</tr>
<tr>
<td>MFCC</td>
<td>0.6036</td>
<td>0.2690</td>
<td>0.2325</td>
<td>0.2690</td>
<td>0.2325</td>
</tr>
<tr>
<td>SMCF</td>
<td>0.6255</td>
<td>0.2821</td>
<td>0.1980</td>
<td>0.2821</td>
<td>0.1980</td>
</tr>
</tbody>
</table>

Summary

A variant of group delay feature derived from the flattened spectrum is introduced in this chapter. The power spectrum is flattened using cepstral smoothing to annihilate the system characteristics. The flattened power spectrum is analyzed using MODGD algorithm followed by DCT, to convert to SMCF features. The potential of the new feature is experimentally evaluated on diverse datasets by two experiments, namely audio indexing task and estimating the number of speakers in mixed speech. The results demonstrate the potential of the new feature in speech/music signal processing and content based information retrieval applications.
CHAPTER 6

Summary and Conclusions

6.1 Summary

The group delay function which is defined as the negative derivative of the unwrapped short-time phase spectrum, can be computed directly from the speech signal, without unwrapping the short-time phase spectrum. The group delay function has been effectively used to extract various source and system parameters. These features have also been gainfully converted to appropriate features for speech synthesis/recognition.

Three tasks in speech/music processing have been addressed using modified group delay functions. They are the predominant melodic pitch estimation in polyphonic music, multipitch estimation in mixed speech and speech/music detection in audio recordings. The features extracted from the computed melodic contour is also used in automatic music genre classification. A new feature is derived from the flattened power spectrum and is effectively used in speech/music discrimination and estimation of number of speakers in mixed speech.

In melody extraction, two paradigms are proposed. In MODGD (Direct), modified group delay analysis is performed directly on the music signal to compute predominant melodic pitch. In this approach, the proper selection of cepstral window emphasizes the peak corresponding to pitch in the group delay spectrum. The search space in this algorithm is limited by the first formant frequency. For high pitched voices, the method ends-up with erroneous pitch estimate, where a low formant can be confused for the pitch frequency. To overcome the stringent constraints in the first approach, group delay analysis on flattened spectrum MODGD (Source) is proposed. The spectrum flattening annihilates the system characteristics and MODGD analysis is performed on the flattened spectrum. Transient analysis is incorporated to capture the dynamic variation of melody. The consistency is ensured by dynamic programming. Median filtering is performed to get a refined melodic pitch contour in the post-processing phase. Automatic music genre classification using fusion of melodic features and group delay features
is also addressed. Results show the promise of the complementary nature of melodic feature in automatic music genre classification.

In the second part, the thesis addresses the problem of multipitch estimation using group delay function in clean and noisy conditions. Iterative estimation and cancellation technique is adopted in the proposed method. The predominant pitch is estimated and the estimated component along with harmonics are cancelled out from the flattened spectrum using comb filters. Pitch grouping and median filtering is performed in the post-processing stage to compute the individual pitch trajectories.

In the third part, a variant of group delay feature is introduced. The new feature representation is derived from the flattened power spectrum after decorrelating using discrete cosine transform. This feature is used in two tasks, namely, estimation of number of speakers in mixed speech and speech/music detection.

The major contributions of the thesis are as follows:

- Two algorithms based on modified group delay functions are proposed for melodic pitch estimation in polyphonic music.
- The complementary nature of high level features derived from melodic pitch contour is studied extensively for automatic music genre classification.
- A new approach based on modified group delay processing, exploiting the high resolution property is proposed for two-pitch tracking.
- A variant of group delay feature is introduced and two applications of the new feature, namely estimating number of speaker in mixed speech and speech/music detection are addressed.

6.2 Criticism

The following are the criticisms regarding the work presented herein.

- We have observed significant gap in raw pitch accuracy and raw chroma accuracy in many audio files. Efficient octave error minimization techniques should be incorporated in the post processing phase to reduce the chroma error in melodic pitch estimation.
- In multipitch estimation, pitch trajectory estimation in the case of cross trajectory is not addressed. Since dynamic programming fails at the point of pitch crossing, methods utilizing speaker characteristics should be incorporated to track the
pitch trajectories. In addition, pitch estimation for more than two speakers is ad-
dressed, but for tracking, speaker-specific characteristics should also be employed in multipitch environment.
Mel-frequency cepstral coefficients

Mel Frequency Cepstral Coefficients (MFCCs) are a feature widely used in automatic speech and speaker recognition. It was introduced by Davis and Mermelstein in the 1980’s, and have been state-of-the-art ever since. According to psychophysical studies, human perception of the frequency content of sound follows subjectively defined non-linear scale called the mel scale. The subjective pitch in mels, $f_{\text{mel}}$ corresponding to $f$ can be computed by,

$$f_{\text{mel}} = 2595 \log_{10} \left( 1 + \frac{f}{700} \right)$$  \hspace{1cm} (A.1)

This leads to the definition of MFCC, a baseline acoustic feature set for speech and speaker recognition applications.

Let $\{y[n]\}_{n=1}^{N}$ represent a frame of speech that is pre-emphasized and Hamming windowed. First $y[n]$ is converted into frequency domain by an $M_s$-point (DFT) which leads to the energy spectrum.

$$|Y[k]|^2 = \left| \sum_{n=1}^{N} y[n].e^{-j2\pi nk/M_s} \right|^2$$  \hspace{1cm} (A.2)

where $l \leq k \leq M_s$.

A filter bank, uniformly spaced in mel scale is imposed on the spectrum calculated in Equation A.2. The outputs $e_{i=1}^{Q}$ of the Mel-scaled band-pass filters can be calculated by a weighted summation between respective filter response $\psi_i[k]$ and energy spectrum $|Y[k]|^2$ as

$$e_i = \sum_{k=1}^{M_s} |Y[k]|^2 \psi_i[k]$$  \hspace{1cm} (A.3)

Finally, Discrete Cosine Transform (DCT) is taken on the log filter bank energies $\log(e[i])_{i=1}^{Q}$ and the final MFCC coefficients $C_m$ can be written as

$$C_m = \sqrt{\frac{2}{Q}} \sum_{l=0}^{Q-1} \log(e[l + 1]) \cos \left[ m \left( \frac{2l - 1}{2} \right) \frac{\pi}{Q} \right]$$  \hspace{1cm} (A.4)
where $0 \leq m \leq R_{cep} - 1$, and $R_{cep}$ is the desired number of cepstral coefficients. Normally, $Q = 20$ and 10 to 30 cepstral coefficients are taken for speech processing applications. The first coefficient $C_0$ is discarded because it represents spectral energy.
APPENDIX B

Melody extraction vamp plug-in

A graphical user interface (GUI) for automatic melody extraction from polyphonic music is inevitable in many music processing applications. Melody based reference templates required for the searchable database in query by humming systems must be extracted from polyphonic soundtracks.

Publicly available pitch detection interfaces, like PRAAT \(^1\) and the Aubio pitch detector \(^2\) for the Sonic Visualizer program \(^3\) have been designed for monophonic audio and cannot be expected to perform acceptably well on polyphonic audio. However, a few polyphonic transcription tools also exist. The polyphonic transcription VAMP plugin \(^4\) has been designed exclusively for guitar and piano music and outputs MIDI notes. Melodyne’s Direct Note Access (DNA) \(^5\) attempts to isolate simultaneously played musical notes but is not freely available for evaluation. MELODIA (Salamon and Gomez, 2012), is a newly proposed GUI with more user friendly approaches for melody extraction from polyphonic music.

A plug-in is developed based on the proposed method of melody extraction as part of the work. In the proposed method, the music signal is flattened to annihilate system characteristics and the residual signal is analyzed using group delay algorithm to estimate the predominant pitch contour. Plug-in can be used to visualise the prominent melody of an audio recording, using Sonic Visualiser vamp host. It allows the user to load an audio file, extract the melody and visualise its pitch sequences. The output is the melodic contour estimated with hop-size specified. Each row of the output contains a time stamp and the corresponding frequency of the melody in Hertz. Two snapshots of the interface are provided in Figures B.1 and B.2. It consists of a waveform viewer, a spectrogram viewer, a menu bar, controls for audio viewing, scrolling and playback and a parameter window.

\(^1\)http://www.praat.org
\(^2\)http://aubio.org
\(^3\)http://www.sonicvisualiser.org
\(^4\)http://isophonics.net/QMVampPlugins
\(^5\)http://www.celemony.com/cms/index.php
Figure B.1: Selection of plug-in on Sonic Visualiser
Figure B.2: Melodic sequences of proposed algorithm along with MELODIA reference.
REFERENCES


LIST OF PUBLICATIONS


5. Rajeev Rajan and Hema A. Murthy, “Melodic pitch extraction from music signals using modified group delay functions”, In Procee. of Communications (NCC), National Conference on, pp 1–5, Indian Institute of Technology, Delhi, February 2013.