WHO IS SPEAKING?
MALE OR FEMALE

A dissertation submitted to the University of Manchester
for the degree of Master of Science
in the Faculty of Engineering and Physical Sciences

2013

By
Hassam Ullah Sheikh
School of Computer Science
3.5 Voice Activity Detector ........................................ 28
   3.5.1 The Empirical Mode Decomposition Method ............ 29
   3.5.2 The Hilbert Spectrum Analysis .......................... 29
   3.5.3 Voice Activity Detection ................................. 31

4 Gender Identification Systems .................................. 32
   4.1 Acoustic Features .......................................... 32
      4.1.1 Mel Frequency Cepstral Coefficients (MFCC) ........ 33
      4.1.2 Shifted Delta Cepstral (SDC) ......................... 34
      4.1.3 Pitch Extraction Method ............................. 36
   4.2 Pitch Based Models ....................................... 36
   4.3 Models based on Acoustic Features ....................... 37
   4.4 Fused Models ............................................. 38

5 Learning Techniques for Gender Identification ............... 39
   5.1 Overview .................................................. 39
   5.2 Adaboost .................................................. 40
   5.3 Gaussian Mixture Model (GMM) ............................ 42
      5.3.1 GMM Training ........................................ 43
      5.3.2 GMM Testing ......................................... 44
   5.4 Decision Making .......................................... 45
   5.5 Likelihood Ratio ......................................... 45
   5.6 Universal Background Model .............................. 45
      5.6.1 UBM Training ........................................ 46

6 System Design and Implementation ............................. 47
   6.1 Toolboxes ................................................. 47
      6.1.1 Signal Processing Toolbox ........................... 47
      6.1.2 Machine Learning Toolbox ........................... 48
   6.2 System Design ............................................ 48
      6.2.1 Requirement ......................................... 48
      6.2.2 Initial Approach .................................... 48
6.2.3 Algorithm .................................................. 49
6.2.4 Feature Selection ......................................... 54
6.3 Experiments and Results .................................. 56
   6.3.1 Pitch Based Models .................................... 56
   6.3.2 Models Based on Acoustic Features ................. 60
   6.3.3 Fused Model ........................................... 63
   6.3.4 YouTube Videos ....................................... 66

7 Conclusion ....................................................... 68
   7.1 Summary .................................................... 68
   7.2 Future Recommendation .................................. 69

Bibliography .......................................................... 70

A Appendix ............................................................ 76

Word Count: 14,045
## List of Tables

6.1 Results from pitch based model trained with 1 male and 1 female speaker .................................................. 57
6.2 Results from pitch based model trained with 9 male and 1 female speakers ...................................................... 58
6.3 Results from pitch based model trained with 1 male and 9 female speaker ....................................................... 58
6.4 Results from pitch based model trained with 8 male and 8 female speakers ....................................................... 59
6.5 Results from MFCC model trained using 8 GMM components ................................................................. 61
6.6 Results from MFCC model trained using 16 GMM components ................................................................. 61
6.7 Results from MFCC model trained using 32 GMM components ................................................................. 61
6.8 Results from SDC model trained using 8 GMM components ................................................................. 62
6.9 Results from SDC model trained using 16 GMM components ................................................................. 63
6.10 Results from SDC model trained using 32 GMM components ................................................................. 63
6.11 Results from fused model trained using 8 GMM components on SDC features ................................................. 64
6.12 Results from fused model trained using 16 GMM components on SDC features ................................................. 64
6.13 Results from fused model trained using 32 GMM components on SDC features ................................................. 65
6.14 Results from acoustic and fused models tested on large amount of data ..................................................... 65
6.15 Accuracy of all the models that were tested on YouTube Videos .......................................................... 66
List of Figures

2.1 Mechanism of the human speech system which is representing the underlying phenomenon of speech generation and speech understanding. The grey boxes are representing computer systems for natural language processing [HAH01] .......................... 16
2.2 Conversion of analogue signal to digital signal. Red lines show the digital value of the analogue signal .............................. 17
2.3 DFT applied to a speech signal ........................................ 19
2.4 DCT applied to a speech signal ........................................ 20
2.5 Sampling of a Continuous Signal ....................................... 21
2.6 Hamming window effect .................................................. 22
3.1 A noisy speech signal [Vat12] .......................................... 26
3.2 A clean speech signal [Vat12] .......................................... 27
3.3 VAD applied to noisy speech [SZ12] ................................. 30
4.1 A block diagram of gender identification model ................. 33
4.2 A block diagram of MFCC computation [Vat12] ............... 33
4.3 Mel frequency scale [Vat12] ............................................ 34
4.4 Graph of Mel filterbank of 24 filters [Vat12] ..................... 34
4.5 Computational model of SDC [TcSKD02] .......................... 35
4.6 Plot of male and female pitch [Sun00] .............................. 35
4.7 Block diagram of gender identification model trained using MFCC [Sun00] .......................................................... 36
4.8 A fused gender identification model [Sun00] ...................... 37
Abstract

The aim of this project was to create a gender identification system that can be used to identify the gender of the speaker. In this dissertation I have explained the signal processing background such as Fourier transforms and DCT etc. that was needed to understand the underlying signal processing happening in digital devices. Apart from that I also investigated the different classification techniques such as Adaboost and Gaussian Mixture Models and different types of methods such as Fusion method, acoustic methods and pitch methods used in gender identification.

From this perspective I have implemented 3 types of models (4 Models) that are explained in the literature and introducing a new method for gender recognition that uses SDC feature with pitch to identify the gender. All models were tested and trained on the same amount of speech. The SDC and SDC fused model gave satisfactory results on Voxforge dataset. Finally I tested the acoustic and fused models on YouTube video which gave almost 90% accuracy. The results of my implementations are shown in chapter 6.
Declaration

No portion of the work referred to in this dissertation has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.
Copyright

i. The author of this report (including any appendices and/or schedules to this report) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=487), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see http:
//www.manchester.ac.uk/library/aboutus/regulations) and in The University’s policy on presentation of Theses
Acknowledgements

I would like to thank my supervisor Dr Ke Chen for his support and guidance throughout the duration of this project without whom this dissertation would never be a reality. Doing this project was a great experience and I learned a lot.

I would also like to thank my dear friend Mr Manav Priyan Olikara for his help and advice regarding this topic and Ali Zia who helped me in generating the graphics for this project.

I would finally like to thank my friends and family for their support.
Chapter 1

Introduction

As the significance of the computers in our daily life is getting popular, the interaction between human and machine is becoming more important day by day. The desire of humans to communicate with machines in a natural way has led to the evolution of natural language processing. As the advancements in this field are happening, it is likely that voice interaction systems will replace the standard keyboards in near the future. Today if we look in the technology market around us we have some really state of the art technologies like Microsoft® Kinect and Apple® SIRI which performs really well. But every speech system that is available today has its own drawbacks and continuous work is being done to increase the performance of such systems. To increase the performance of speech systems pre-processing like gender recognition and language identification is required.

This MSc project focuses on automatic gender identification system using speech. Identification of gender using the speech of the speaker concerns in detecting that the spoken speech is of male or female speaker. Automatic Gender Identification (AGI) via speech has several applications in the field of natural language processing. In [AH96] has shown that gender dependant speech recognition models are more accurate than gender independent models. Google’s latest speech recognition system that can be seen in android devices and Google Glass initially finds the gender of the speaker before performing the speech recognition for search. The result of its speech recognition accuracy is exceptionally high as compared to their previous
speech recognition system which was an unisex model. Recently a company has launched its “Kinect” based online fitting room that determines the gender of the person using it speech to offer him clothes. In the context of multimedia indexing gender recognition can considerably decrease the search space up to half [HC].

Automatic gender recognition itself is a complex task and it has its own problems and limitations, until now no gender recognition system exists which can work on real time environment with 100% accuracy. As in a real world environment or in the case of multimedia indexing many acoustic conditions exist like noisy speech, compressed speech, silence, speech on the telephone, different languages and so on which significantly reduces the performance of a general gender identification system. So ideally a system is required which can give acceptable performance under previously described acoustic conditions.

In general, there are three main approaches to building an automatic gender identification system: The first approach uses pitch as a discriminating factor and use labelled data to identify the gender of the speaker. The second approach deals with acoustic features like MFCC and unlabelled data to identify the gender. In this approach relevant features are extracted then the model is trained. In this case generally a GMM is trained for each gender and results from one model are subtracted from the other model to find the gender. The third approach is quite commonly used after year 2005, in which pitch models are combined with acoustic models to form a fused model.

The dissertation is organised as follows: The first chapter presents the challenge of gender identification system. The second chapter presents necessary knowledge of signal processing. Then each chapter describes a step of a gender identification system: including speech enhancement techniques to reduce background noise, feature extraction, gender modelling methods and different decision making techniques. Finally, the last chapter presents the implementations done for this project and results obtained from testing different models on a large set of speakers and YouTube videos.
Chapter 2

Background

To understand the gender identification process using speech, we first need to understand the structure of speech. This chapter includes human speech and what is the basic difference between female and male voice.

2.1 Speech

Spoken language or human speech is the natural form of human communication which requires the use of voice. In terms of Linguistics human speech is a form of sound wave which is produced by the lungs and it is given uniqueness by tongue, lips and jaws [HAH01]

2.1.1 Speech Signal

Speech is produced when the air pressure generated by lungs reaches the vocal cords. Then speech begins to resonate in the nasal cavities according to the position of lips, tongue and other organs in the mouth. In terms of signal processing speech signal is an analogue signal which is the convolution of the source $e[n]$ and a filter $h[n]$ which can be seen in equation 2.1 where lungs can be modelled as the source
Figure 2.1: Mechanism of the human speech system which is representing the underlying phenomenon of speech generation and speech understanding. The grey boxes are representing computer systems for natural language processing [HAH01]

\[ x[n] = e[n] \ast h[n] = \sum_{k=-\infty}^{\infty} e[k]h[n - k] \]  

(2.1)

where \( x[n] \) is the speech signal.

**Human Speech Frequency**

The frequency range that is the part of the audio range is 300Hz to 3400Hz which means that human speech lies in this range [Tit00a]. On the other hand the sound ranges i.e. the frequency range between humans can hear any sound is between 20 Hz to 20,000Hz. Beyond the region of 20,000Hz the region of ultrasonic comes which, humans are unable to hear.
2.2. SPEECH SIGNAL PROCESSING

Fundamental Frequency (Pitch)

Generally fundamental frequency is defined as the minimum frequency of the periodic waveform. The fundamental frequency or usually known as pitch in terms of natural language processing is the biggest discriminating factor between a male and a female speech. A typical male adult has a fundamental frequency between 85Hz to 180Hz and an adult female has a fundamental frequency in the range of 165Hz to 225Hz [BO00].

2.2 Speech Signal Processing

From [HAH01] we know that speech is an analogue signal but unfortunately today’s computers work with digital signals so speech is saved in digital form in computers. When speech is converted to digital form, it loses some of the data so accurate representation of analogue signal into digital signal is required. A conversion of analogue signal can be seen in figure 2.2.
2.2.1 Fourier Transform

According to Joseph Fourier, any signal can be represented as a linear combination of sinusoids which means that the Fourier transform can be described as transforming a function of time $f(t)$ into a function of frequency $F(\omega)$. This can be shown as

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i \omega t} dt$$  \hspace{1cm} (2.2)

There exist different types of Fourier transforms but most famous are

1. Continuous Time Fourier Transform
2. Continuous Fourier Transform
3. Discrete Fourier Transform
4. Discrete Time Fourier Transform

In an automatic gender recognition system only discrete Fourier transform is required so only that will be explained.

**Discrete Fourier Transform**

For any periodic signal $x[t]$ the discrete Fourier transform can be defined as

$$X(\omega) = \sum_{t=0}^{T-1} x[t] e^{-2\pi i \omega t}$$  \hspace{1cm} (2.3)

A discrete Fourier transform applied to a signal can be seen in figure 2.3

2.2.2 Discrete Cosine Transform

Discrete Cosine Transform commonly known as DCT is similar to DFT. DCT is used to transform a finite sequence of data points into sum of different sinusoids vibrating at different frequencies. DCT is usually used for compression of images and sound where the lower number of higher frequency components can be discarded which means that the transformed signal is mostly comprised of lower frequencies.
2.2. SPEECH SIGNAL PROCESSING

Figure 2.3: DFT applied to a speech signal

thus majority of information can be found in first coefficient. More information about the usage of the DCT in speech processing can be found in [MAMG11]. Mathematically DCT can be defined as

\[ X_T[k] = \sum_{n=0}^{T-1} x_T[n] \cos \left( \frac{k\pi}{T} \left( n + \frac{1}{2} \right) \right), \quad k \in [0, T-1] \]  

(2.4)

where \( X_T[k] \) is the \( k \)th coefficient. DCT applied to a speech signal can be seen in figure 2.4

2.2.3 Digital Filters

Digital filters are mathematical models that are applied to a signal to remove some components of that signal or to enhance some aspects of the signal. In natural language processing widely used filters are low pass filters, band pass filters and high pass filters [APHA96].

Low Pass Filter

A low pass filter is used to discard the frequencies higher than the cut-off frequency in a speech signal.
CHAPTER 2. BACKGROUND

**Figure 2.4: DCT applied to a speech signal**

**High Pass Filter**

A high pass filter is used to discard the frequencies lower than the cut-off frequency in a speech signal.

**Band Pass Filter**

The band pass filter allows a certain range of frequencies to pass and discard all the frequencies that are higher or lower than the cut-off frequencies.

**Sampling**

Human speech signal is naturally a analogue signal but to perform any computational tasks on the speech signal, it should be converted to a digital form. In signal processing, sampling means to convert a continuous time signal to discrete time signal by looking at in regular intervals of time. The regular interval of time is generally called the sampling interval which is the reciprocal of the sampling frequency and is denoted by $T_s$. Sampling frequency, generally known as the sampling rate is defined as number discrete samples taken from a signal in one second and is denoted by $f_s = \frac{1}{T_s}$. The higher the sampling frequency is the better the digital signal is as more information was captured and less information was lost. Usually
in speech processing 44 KHz is considered to be a good sampling rate which means that 44000 samples are taken from 1 second of speech.

Quantization

In digital signal processing quantization is the process of mapping a continuous range of very large values to a smaller set of discrete or integer values. The error that is induced because of the loss of the information during mapping is called quantization error. Quantization is the used in analogue-to-digital converters for converting discrete signals to digital signals using a quantization level specified in bits. As loss of information during quantization is irreversible, it is a good practise to set the quantization level to higher bits. A good quality compact disc is sampled at 44.1 KHz with a quantization level of 16 bits which can give 65,536 possible values per sample.

![Analogue Waveform](image.png)

Figure 2.5: Sampling of a Continuous Signal

2.2.4 Nyquist Shannon Sampling Theorem

Nyquist Shannon sampling theorem more generally known as Nyquist sampling theorem states that
"If a function $x(t)$ contains no frequencies higher than $B$ hertz, it is completely determined by giving its ordinates at a series of points spaced $\frac{1}{2B}$ seconds apart [Wik13b]."

Which actually means that to reconstruct a continuous signal from its digital form, the sampling rate $f_s$ should be twice as greater than the bandwidth $B$ of the signal.

$$f_s > 2B \tag{2.5}$$

### 2.2.5 Window Functions

In signal processing, window function is defined as a mathematical function who value is zero outside the given interval. As a person is talking, sound produced by it changes very quickly so to study every change/segment of speech, it is divided into many frames with help of a window function.

![Figure 2.6: Hamming window effect](image)

Most famous window functions are rectangular and Hamming windows. Rectangular function is always constant inside the given interval and always zero outside the interval. Rectangular window is a constant inside the interval which means that the changes around the edges are abrupt so to decrease this abrupt effect, Hamming
window is used which is generalized by equation.

\[
h_N[n] = \begin{cases} 
\alpha - \beta \cos \left( \frac{2\pi n}{N-1} \right) & 0 \leq n < N \\
0 & \text{otherwise}
\end{cases}
\]  

(2.6)

A hamming window effect can be seen in figure 2.6. In MATLAB’s signal processing toolbox hamming window function is available which can be used by giving command ‘hamming(Signal)’
Chapter 3

Speech Enhancement

Automatic gender identification systems trained on clean speech under lab environments degrade in performance in real world conditions due to the additive environmental sounds such as noise and music, additionally there is a natural pause in human speech which is known as silence is also considered as an additive environment sound. Systems whose performance does not degrade in a real world environment are called robust systems.

Several researchers are trying to design an auditory system which can mimic the auditory system of human beings as it is robust to environmental changes [Ghi87]. Before training a speech based system, speech enhancement pre-processing has to be done so that a good quality of speech can be extracted from speech recorded in a real world environment. In this chapter some speech enhancement techniques will be discussed.

3.1 Signal to Noise Ratio

Signal to noise ratio is a measure that is used to find the quality of the input signal according to the background noise in the signal. Signal to noise ratio is usually denoted as SNR. Mathematically SNR can be written as

$$SNR = \frac{P_{Speech}}{P_{Noise}}$$  \hspace{1cm} (3.1)
3.2. SPECTRAL SUBTRACTION

Where $P$ is the power of the input signal

$$P = \sqrt{\frac{1}{T} \sum_{n=0}^{T-1} x[n]^2} \quad (3.2)$$

Where $x[n]$ is the input signal and $T$ is the time period of the signal. According to equation 3.1, higher the SNR is, the better the speech quality is in the input signal. "Voicebox" created by [Bro11] is a MATLAB toolbox that has all speech processing functions, in Voicebox SNR can be calculated using the function "snrseg"

3.2 Spectral Subtraction

In real world speech signal, the basic assumption is that there is a clean speech signal $s[n]$ which is corrupted by some additive environmental sound $y[n]$ so the real world speech signal can be written as

$$x[n] = s[n] + y[n] \quad (3.3)$$

As $s[n]$ and $y[n]$ are more independent than the power spectrum of $x[n]$ can be written as the sum power spectra of $s[n]$ and $y[n]$.

$$|X(f)|^2 = |S(f)|^2 + |Y(f)|^2 \quad (3.4)$$

Although we don’t know the power spectrum of the additive noise $|Y(f)|^2$, but we can estimate by averaging the $|X(f)|^2$ over $M$ frames [HAH01]

$$|\hat{Y}(f)|^2 = \frac{1}{M} \sum_{i=0}^{M-1} |X_i(f)|^2 \quad (3.5)$$

By using equation 3.4 and 3.5 we can estimate $|S(f)|^2$

$$|\hat{S}(f)|^2 = |X(f)|^2 - |\hat{Y}(f)|^2 = |\hat{Y}(f)|^2 \left(1 - \frac{1}{SNR(f)}\right) \quad (3.6)$$
CHAPTER 3. SPEECH ENHANCEMENT

3.3 Cepstral Mean Normalization

Same speech recorded in the same environment from different microphones may give different types of sound because every microphone has its own transfer function. Even the same microphones can a varying transfer function depending upon the distance of the user from the microphone or the acoustic conditions of the room. In this section a powerful technique is described which is used to handle such type of distortions. This technique is called Cepstral Mean Normalization which is described below. Given an input signal $x[n]$, we calculate its cepstrum features by doing short time analysis which gives us a set of cepstral features of $T$ vectors.
3.3. **CEPSRAL MEAN NORMALIZATION**

![Figure 3.2: A clean speech signal [Vat12](image)](image)

The Cepstral Mean Normalization [Ata74] consists of subtracting the mean \( \bar{x} \) from every vector \( x_i \) of the dataset to get normalized cepstrum vectors \( \hat{x}_i \). Consider a speech signal \( y[n] \) which is obtained by filtering signal \( x[n] \) using a filter \( h \), then we can compute the cepstral features \( Y \). The mean of \( Y \) can be calculated by

\[
\bar{y} = \frac{1}{T} \sum_{t=0}^{T-1} y_t = \frac{1}{T} \sum_{t=0}^{T-1} (x_t + h) = \bar{x} + h
\]  

(3.8)

\[
\hat{y}_t = y_t - \bar{y}_t = \hat{x}_t
\]  

(3.9)

CMN is performed on every utterance of both training and testing data. Experiments have shown that CMN can provide 30% less error rate thus providing robustness to
3.4 RASTA Filtering

Relative Spectra commonly known as RASTA [HM94] was designed to support the PLP pre-processing to remove slow channel variations in the speech signal thus reducing the noise. RASTA uses band pass filters to remove slow variations in the signal which in turn leads to the removal of the magnitudes which are constant. RASTA filter can be written as

\[ H(z) = 0.1z^4 \cdot \frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{1 - 0.98z^{-1}} \]  

(3.10)

3.5 Voice Activity Detector

When a person is speaking, there are natural pauses in his speech or some other non-speech sounds, these pauses are also known as silence. The data of the speech signal is corrupted by silence or non-speech sounds. For avoiding signal from being corrupted these non-speech frames should be removed from the signal so that correct relevant features can be extracted from the speech signal.

Voice Activity Detectors are used to identify non speech frames from a given input audio signal. Voice detection is very important pre-processing part of speech processing techniques like language identification and speech recognition. Applications of VAD include lowering the bit rate of speech coders, controlling the estimation routines used in echo cancellation and noise reduction methods and controlling automatic gain routines. VAD are generally applied on the speech signal right after the speech signal is digitized.

VAD system can be made using Hilbert-Huang Transform (HHT) framework which can extract intrinsic mode functions using empirical mode decomposition [SZ12]
3.5. **VOICE ACTIVITY DETECTOR**

3.5.1 **The Empirical Mode Decomposition Method**

EMD assumes that any data that consists of different IMFs can be represented as a linear combination of those IMFs and some remaining factor [SZ12]. Given an input signal \( x(t) \) the EMD algorithm according to [HSL\(^+\)98] is described below.

1. Find all local minima and maxima of \( x(t) \)

2. Generate lower and upper envelops of \( x(t) \) via cubic spline interpolation using all the minima and maxima respectively.

3. The average of both envelops is designated as \( m_1 \) and the first IMF candidate \( h_1 \) is the difference between the average and the input signal, \( h_1 = x(t) - m_1 \)

4. Apply the sifting process which is replacing \( x(t) \) with \( h_1 \) and repeating steps from 1 to 3. Then \( h_{11} = h_1 - m_{11} \)

5. Repeat the sifting process for \( k \) times which is \( h_{1k} = h_{1(k-1)} - m_{1k} \) until \( h_{1k} \) satisfies the constraints explained in [HSL\(^+\)98]. Then the first IMF \( c_1 = h_{1k} \) and remaining factor \( r_1 = x(t) - c_1 \)

6. Repeat steps from 1 to 5 until a predefined criterion is met. Then the original signal can be written as

\[
x(t) = \sum_{j=1}^{N} c_j + r_n
\]

3.5.2 **The Hilbert Spectrum Analysis**

Now we have \( N \) order IMFs we can calculate the Hilbert transform of these IMFs. The complex conjugate \( c_j^\prime(t) \) of the \( j-th \) order IMF \( c_j(t) \) can be calculated using equating 3.11 [SZ12]

\[
c_j^\prime(t) = H\{c_j(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{c_j(\tau)}{t - \tau} d\tau
\]

(3.11)
The analytic function is defined as

\[ z_j(t) = c_j(t) + ic'_j(t) \]  \hspace{1cm} (3.12)

The instantaneous amplitude and phase function \( a_j(t) \) and \( \phi_j(t) \) can be found out using equation 3.13 and 3.14 respectively

\[ a_j(t) = \sqrt{c_j^2(t) + c'_j(t)} \]  \hspace{1cm} (3.13)

\[ \phi_j(t) = \arctan \left( \frac{c'_j(t)}{c_j(t)} \right) \]  \hspace{1cm} (3.14)

Instantaneous frequency can be found out by

\[ f_j(t) = \frac{1}{2\pi} \frac{d\phi_j(t)}{dt} \]  \hspace{1cm} (3.15)
This plays an important role in making a VAD system

### 3.5.3 Voice Activity Detection

The Instantaneous Frequency Average (IFA) and weighted sum of IFA of each IMF provides a good indication to detect voice activity in a sound signal [SZ12]. IFA and WIFA can be represented as

\[
p_{j,l} = \frac{T_f}{t=0} f_{j,l}(t)/T_f \quad \text{and} \quad P_l = \sum_{j=2}^{N} 2^{j-1} p_{j,l}
\]  

(3.16)

Where \(T_f\) is the duration of each frame and \(f_{j,l}\) is the IFA of the \(j-th\) IMF component of the \(l-th\) frame. For every frame if value \(P_l\) exceeds a particular threshold value, that frame will be marked as a speech.
Chapter 4

Gender Identification Systems

Now all the basic understandings of human speech and speech enhancement methods have been reviewed, we can now see how gender identification system works. In this chapter I will explain the training and testing process of gender identification systems including the features used to make such systems. This chapter also describes the different types of gender identification system that are the result of previous research and how they used speech features to build AGI systems.

4.1 Acoustic Features

Acoustic features can be defined as the acoustic property of sound that is used to find distinctive features of a class of speech sounds. In natural language processing one feature named Mel-Frequency Cepstral Coefficients (MFCC) dominates all other features. MFCCs are not only used in gender identification but they are also used in speech recognition and language identification systems. MFCC has been used by [SGS12] and [DBK12] to achieve high accuracy. A gender identification model can be seen in figure 4.1
4.1. ACOUSTIC FEATURES

4.1.1 Mel Frequency Cepstral Coefficients (MFCC)

MFCC are coefficients that collectively form Mel frequency cepstrum which is a power spectrum of a short window of a speech signal i.e. 25 milliseconds etc. MFCC tries to represent the shape of the vocal tract using the short term power spectrum thus trying to approximate human auditory system responses. As the MFCC represents a short term power spectrum so they are considered to be short term features. The block diagram of MFCC computation can be seen in 4.2 A better set of features is shifted delta cepstral coefficients which are derived from MFCC and have proven to be better than MFCC in natural language processing tasks. Following the block diagram, the MFCC computation according to [Wik13a] can be described as

1. Take DFT of the sampled window
2. For each frame calculate the periodogram estimate of the power spectrum
3. Take log of all the powers at every frequency
4. Take DCT on all the Log Mel powers computed in step 3
5. MFCC are the amplitudes of the spectrum that is created by the taking the DCTs in step 4
4.1.2 Shifted Delta Cepstral (SDC)

Shifted Delta Cepstral coefficients are considered to be long term features which are derived using MFCCs, SDC has 4 parameters "$N - d - p - k$" and according to [TcSKD02] these parameters are described as "where N is the number of cepstral coefficients computed at each frame, d represents the time advance and delay for the delta computation, k is the number of blocks whose delta coefficients are concatenated to form the final feature vector, and P is the time shift between consecutive blocks". For every frame $t$, the SDC vector can be written as

$$SDC(t) = \begin{pmatrix} \Delta c(t,0) \\ \Delta c(t,1) \\ \Delta c(t,2) \\ \vdots \\ \Delta(t,k-1) \end{pmatrix}$$  \hspace{1cm} (4.1)
4.1. ACoustIC FEATURES

For every frame $t$ SDC vector uses $Kp$ consecutive frames of cepstral coefficients that is why SDC are considered as long term features.
4.1.3 Pitch Extraction Method

Pitch or fundamental frequency is considered to be a most basic discriminating factor between male and female voice. A typical male adult has a fundamental frequency between 85Hz to 180Hz and an adult female has a fundamental frequency in the range of 165Hz to 225Hz [BO00]. Pitch determination is still considered to be a difficult task because of the common errors like pitch doubling and pitch halving that happens because of the presence of an alternate pulse cycle which shows the instability of human vocal fold system [Tit00b]

Pitch determination algorithm proposed by [Sun00] based on Sub harmonic to Harmonic ratio uses log frequency scale and spectrum shifting technique to obtain the amplitude summation of harmonics and sub harmonics respectively [Sun00]. The pitch of normal speech and the alternate cycle can be determined by comparing the amplitude ratio of harmonics and sub harmonics with the pitch perception result [Sun00]

![Figure 4.7: Block diagram of gender identification model trained using MFCC [Sun00]](image)

4.2 Pitch Based Models

Pitch based models are those gender identification models that use only pitch as a feature to determine the gender of the speaker. This type of model is implemented
in Microsoft Kinect. These systems find a thresholding value during the training. When a new speech comes, its pitch value is compared to the threshold value, if the value is less than the threshold value than the speaker is male and if the value is greater than the threshold value than the speaker is female. Usually the threshold value is set near 200Hz. Pitch based models only work well for clean speech. Their performance degrades significantly when used in real world environments.

4.3 Models based on Acoustic Features

Acoustic models are those gender identification models that use acoustic features like MFCC and SDC for gender identification. MFCC has been used by [DBK12] and [RN11] for gender identification. Most of the authors have used Gaussian Mixture Model as a classifier. As GMM is generative model a separate GMM is trained for each class. When a new speaker comes, its value is given to both trained models. The resultant value of female model is subtracted from the resultant value of a male model. If the answer is more than 0, then the speaker is male else it is female. Mathematically this step can be written as

\[ S_G = S_{male} - S_{Female} \] (4.2)

Figure 4.8: A fused gender identification model [Sun00]
4.4 Fused Models

Until 2005 most of the research community was just using the acoustic features like MFCC or only pitch separately for gender identification tasks. After 2005, researchers have started to use acoustic features and pitch together to achieve higher accuracy in gender identification [TYZ06], [DC11] [KH10] and [IKJGY10] have used fused systems to achieve more than 90% accuracy for gender identification. [Sun00] has used the linear normalization method to combine the score from acoustic analysis that used MFCC and score from pitch estimation. [Sun00] method can be mathematically can be written as

\[ S = \frac{(S_{mfcc} - \lambda_{mfcc})}{\alpha} + \frac{(\lambda_{pitch} - S_{pitch})}{(100 \times p)} \] (4.3)

If S is greater than 0 than the speaker is male else it is female. Recently [IKJGY10] has used score level fusion by Adaboost for speaker gender recognition by fusing biometric systems and soft biometric traits.

![Figure 4.9: A adaboost score fusion model [IKJGY10]](image)
Chapter 5

Learning Techniques for Gender Identification

As described that all gender identification systems need a classification model to distinguish between male and female speaker. Some commonly used classifiers are explained in this chapter

5.1 Overview

Commonly used classifiers in the field of automatic gender identification are Support Vector Machines (SVM) and Gaussian Mixture Models commonly known as GMM. SVM by [BGV92] is a supervised learning algorithm which trains a model that is used to classify between different classes. The main idea behind SVM is "find a linear separating hyper plane with the maximal margin in this space" [Hsu10]. This linear hyper plane is used to do classification between different genders provided that the acoustic features are linearly separable [CSRS06]. But in most cases these features are non-linearly separable but SVM allows projecting these features into higher dimensional space where the separating hyper plane is calculated.

As SVM is a linear classifier it can be expressed in the form of

$$w'x + b = 0 \quad (5.1)$$
CHAPTER 5. LEARNING TECHNIQUES FOR GENDER IDENTIFICATION

Anything above the decision will be labelled as class 1 and anything below the decision boundary will be labelled as class -1 or 0.

GMM is an unsupervised learning algorithm which is considered to be best classic parametric method for gender identification because of the fact that Gaussian components can represent gender related information effectively [RN11]. GMM has been explained in more detail later in this chapter.

Recently [IKJGY10] has used score level fusion by Adaboost for speaker gender recognition by fusing biometric systems and soft biometric traits.

\[ K(x_i, x_j) = \phi^\theta(x_i)\phi(x_j) \]

Figure 5.1: Optimal decision boundary between two classes

5.2 Adaboost

In 1990 [Sch90] proposed a method to increase the performance of weak learning algorithms. After improvements made by [Fre90] and further expansion was made in 1996 by [Fre96], AdaBoost (Adaptive Boosting) was introduced by [SF95].
Algorithm 1 Adaboost(training sets $S$ of size $m$, Inducer $I$, integer $T$ (number of trials)).

$S' = S$ with instance weights assigned to be 1

for $i = 1$ to $T$ do

$C_i = I(S')$

$\epsilon_i = \frac{1}{m} \sum_{x_j \in S': C_i(x_j) \neq y_j} \text{weight}(x)$

if $\epsilon_i > 1/2$ then

set $S'$ to a bootstrap sample from $S$ with weight 1 for every instance and go to step 3 (this step is limited to 25 times after which we exit the loop)

end if

$\beta_i = \frac{\epsilon_i}{1 - \epsilon_i}$

For-each $x_j \in S'$, if $C_i(x_j) = y_j$ then weight($x_j$) = weight($x_j$) · $\beta_i$

Normalize the weights of instances so the total weight of $S'$ is $m$.

end for

$C^*(x) = \arg\max_{y \in Y} \sum_{i : C_i(x) = y} \log \frac{1}{\beta_i}$

Output Classifier $C^*$

Bagging and Adaboost just share only one common feature that both generate a set of classifiers and answer is chosen by voting [BK98]. Other than that there is a huge difference in these two algorithms. The Adaboost algorithm is shown in Algorithm 1 ¹, which shows that classifiers are produced in a sequential way rather than parallel as it happens in Bagging.

Adaboost has the ability to change the weight of the training input to the classifier depending upon the previously built classifiers. The goal of Adaboost is to force the set of classifiers to reduce the error rate over different input distribution [BK98]. In Adaboost a number of classifiers are made and in the end a final classifier $C^*$ is created using a voting scheme based on weights.

The complete algorithm for the Adaboost according to [BK98] is shown in Algorithm 1 ²

¹The algorithm described is completely taken from [BK98]
²The given algorithm is completely taken from [BK98]
5.3 Gaussian Mixture Model (GMM)

Gaussian Mixture Model is an unsupervised learning technique that is commonly used in natural language processing for tasks like speech recognition, language identification and gender identification etc. GMM captures and resolve observable ambiguities in data [Jaa]. A $m$ component Gaussian Mixture Model can be written as

$$P(x; \theta) = \sum_{j=1}^{m} p(j)N(x; \mu_j, \Sigma_j)$$

(5.2)

"The parameters $\theta$ include the mixing proportions (prior distribution) $P(j)$, means of component Gaussians $\mu_j$, and covariance $\Sigma_j$. The notation $P(j)$ is a shorthand for $P(j), j = 1, \ldots, m$". [Jaa]

\[ N(x; \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp \left( -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right) \]

(5.3)
5.3. GAUSSIAN MIXTURE MODEL (GMM)

Some Gaussian Mixture Models can be seen in the figure 5.2

5.3.1 GMM Training

The training of Gaussian Mixture Model includes finding the best parameter $\theta$ which can estimate the probability distribution of the training data $X$ efficiently where $D = \{x_1, x_2, \ldots, x_n\}$. The best method to find the most efficient probability distribution of the training data is to maximize the likelihood of $\theta$ using expectation-maximization algorithm. The likelihood for the parameter $\theta$ can written as

$$L(D; \theta) = \prod_{t=1}^{n} p(x_t|\theta) = \prod_{t=1}^{n} \sum_{j=1}^{m} p(j)N(x; \mu_j, \Sigma_j) = \prod_{t=1}^{n} \left( \sum_{j=1}^{m} P(x|\theta_j)P(j) \right) \quad (5.4)$$

Where $x_t$ represent the type of data we have in the distribution. So if we maximize $L(D; \theta)$ we automatically maximize the log of $L$. The log of $L$ can be written as

$$\log L(D; \theta) = \sum_{i=1}^{N} \log \sum_{j=1}^{m} p(j)N(x; \mu_j, \Sigma_j) \quad (5.5)$$

As the maximization is in terms of log then it is a nonlinear maximization. For this reason EM algorithm [DLR77] is used which is an iterative process for maximum likelihood using the estimations made in the last step. The intuition behind EM algorithm is to estimate that in which Gaussian component the data point $x_i$ belongs to. A new latent variable $Z_{ij}$ is introduced which gives 1 when the data point $x_i$ belongs to $j-th$ Gaussian. After introducing $z$, the log likelihood can be written as.

$$\log L(D,Z; \theta) = \sum_{i=1}^{N} \sum_{j=1}^{m} z_{ij} \log \{p(j)N(x; \mu_j, \Sigma_j)\} \quad (5.6)$$

In each iteration 2 steps are performed i.e. Expectation and Maximization which are defined as follows.
CHAPTER 5. LEARNING TECHNIQUES FOR GENDER IDENTIFICATION

Expectation Step

For every iteration $k$ the expectation step can be written as

$$t_{ij}^k = P(j|x_t, \theta^{(l)}) = \frac{P^{(l)}(j)N(x_t; \mu_j, \Sigma_j)}{\sum_{j'=1}^{m} P^{(l)}(j')N(x_t; \mu_{j'}, \Sigma_{j'})} = \frac{P^{(l)}(j)N(x_t; \mu_j, \Sigma_j)}{P(x_t; \theta^{(l)})} \quad (5.7)$$

where $t_{ij}$ is the expected value of $z_{ij}$

Maximization Step

The maximization step is to find parameter $\theta^{k+1}$ which maximizes the function

$$\sum_{i=1}^{N} \sum_{j=1}^{m} t_{ij}^k \log \left\{ p(j)N(x; \mu_j, \Sigma_j) \right\} \quad (5.8)$$

To do the maximization, for every step $k$ all parameters for all $j$ Gaussians should be updated as follows.

$$P^{(k+1)}(j) = \frac{\hat{h}(j)}{h}, \text{where } \hat{h}(j) = \sum_{t=1}^{h} P^{(l)}(j|t) \quad (5.9)$$

$$\mu_j^{(k+1)} = \frac{1}{\hat{h}(j)} \sum_{t=1}^{h} P^{(l)}(j|t)x_t \quad (5.10)$$

$$\Sigma_j^{(k+1)} = \frac{1}{\hat{h}(j)} \sum_{t=1}^{h} P^{(l)}(j|t)(x_t - \mu_j^{(k+1)})(x_t - \mu_j^{(k+1)})^T \quad (5.11)$$

5.3.2 GMM Testing

The testing system of GMM is quite tricky, as GMM is an unsupervised generative model the classification problem becomes a bit complex to solve. For each class of data, a separate GMM is trained and log likelihood for each class is calculated. A back end decision making system is used to compare these different scores generated by each GMM and then a final result is produced.
5.4 Decision Making

Once the different scores from all Gaussian Mixture Models for all the target genders have been computed, we need a system that takes all those scores as an input and makes a decision on recognizing the gender of the spoken utterance. This section explains some decision maker techniques used in Automatic Gender Identification systems.

5.5 Likelihood Ratio

The Likelihood ratio can be defined as the ratio of two same events under different conditions. In gender identification system we can take the maximum score of both genders as the guess gender but this technique is not considered to be effective but if we make a threshold value than we can make a decision about the gender. But this threshold value has be to compare with some meaningful scores rather than the raw scores obtained from the models. The meaningful score/ the likelihood of each gender is given as $y(X|g)$ and the likelihood ratio test can be written as

$$r(X|g) = \frac{y(X|g)}{\sum_{i=g}^G y(X|g)}$$

(5.12)

5.6 Universal Background Model

Universal background model by [RQD00] is a framework that has achieved great success in natural language processing related tasks such as speech recognition, language identification and gender identification. Conceptually UBM is a large Gaussian Mixture Model that adapts to each speaker using maximum a posteriori (MAP) scheme. UBM are large GMMs so it is trained using multiple EM iterations rather than just 1.
5.6.1 UBM Training

As UBM is a large set of Gaussian Mixture Models, the parameters of these Gaussians consist of weights $w_k$, mean $\mu_k$ and covariance matrices $\Sigma_k$. Let us consider that speech is already divided into $J$ speech classes. For training the UBM, the Gaussian posterior is defined as

$$
\gamma_{jk}(t) = \frac{w_k g(x(t)|\mu_k,\Sigma_k)}{\sum_{k=1}^{K} w_k g(x(t)|\mu_k,\Sigma_k)} \tag{5.13}
$$

$$
\gamma_{jk} = \sum_{t=1}^{T} \gamma_{jk}(t), x_{jk} = \sum_{t=1}^{T} \gamma_{jk}(t)x(t), S_{jk} = \sum_{t=1}^{T} \gamma_{jk}(t)x(t)^2 \tag{5.14}
$$

Where $\gamma_{jk}$ is the phone posterior and $x_{jk}$ and $S_{jk}$ are first order statistics [PCV08]. $k$ is the index of the Gaussian in universal background model. The updated parameters after EM will be

$$
w_{jk} = \frac{\gamma_{jk}}{\sum_{j=1}^{J} \gamma_{jk}}, \mu_{jk} = \frac{x_{jk} + \tau \mu_k}{\gamma_{jk} + \tau}, \sum_{jk_{ii}} = \frac{S_{jk} + \tau (\mu_k^2 + \Sigma_k)}{\gamma_{jk} + \tau} - \mu_{jk}^2 \tag{5.15}
$$

As UBM is a background model, it is common to all training classes [RQD00] it needs a large dataset so that it can represent expected data properly. Training UBM is complex. Each gender should have equal number of data so that the data can be represented equally otherwise UBM can be biased to the data that has a large population.
Chapter 6

System Design and Implementation

In previous chapters I have described the different elements of gender identification system ranging from speech enhancement methods to different feature extraction and identification models. In this chapter I will show how I combined all those methods and features described in previous chapters to create 3 different types of gender identification models.

6.1 Toolboxes

The project was done in MATLAB which is high level language commonly used to perform mathematically complex tasks. The two main MATLAB toolboxes I used to develop the system are described below.

6.1.1 Signal Processing Toolbox

Building a gender identification system is mostly a signal processing task. Voicebox toolbox by [Bro11] is a MATLAB toolbox which was used to manipulate audio files, implementing noise reduction and extracting acoustic features like MFCC. To extract SDC features from MFCC codes provided by [Sah12] were used. Finally to extract pitch, the code provided by [Sun02] was used.
6.1.2 Machine Learning Toolbox

Building a gender identification system also involves a great part of machine learning as well. A number of MATLAB toolboxes are available online but LIBSVM toolbox by [CL11] and NETLAB 3.3 toolbox by [NB04] was used to train the SVM model and Gaussian Mixture Models respectively.

6.2 System Design

This section describes all the components of gender identification system used to design a gender identification system. This was a research project so many basic principles of software development and software engineering were ignored.

6.2.1 Requirement

The requirement of my project was to identify the gender of the speaker regardless of the age, language, accent and dialect in real world environments where different additive environmental sounds like silence, background noise and music makes it hard to achieve 100% accuracy. The system should be robust enough to work for variety of different speakers. The first choice for the system was to implement a system in which gender can be recognized while the speaker is speaking into the microphone.

6.2.2 Initial Approach

The fundamental difference between a male and a female speech is pitch. My first approach to build a gender identification system was to create a pitch based model which can identify the gender by extracting the pitch of the speech. To extract the pitch from the speech, code provided by [Sun02] was used. Pitch of every 25 milliseconds of the frame was calculated and mean of all the pitches from all the frames was used as the pitch of speech input. The range of pitch for both genders is between 100 to 300 Hz so all the frequencies below 100 and above 300 was ignored.
For classification of the gender, an SVM was trained using the LIBSVM toolbox by [CL11].

The algorithm used in this system is as follows; first the audio files were loaded in the MATLAB environment, then speech enhancement techniques like spectral subtraction and voice activity detection were applied to remove all the noises and silences respectively using the functions `specsub` and `vodsohn` from voicebox. After that pitch extraction method was implemented in the enhanced audio files. Finally the mean of the pitch extracted for each 25 millisecond frame was taken to determine the pitch of each audio file. Each file served as one data point in the dataset.

The training set was mostly consisted in Librivox recordings of one hour each for each gender at different frequencies ranging from 16 KHz to 48 KHz from Voxforge open source database. Testing data was also taken from the same database. At first only English language speech utterances were taken.

Initially I decided to use the decision stump for this classification but after analysing the data I found out that though the data was one dimensional but it was not linearly separable so I trained a nonlinear SVM model using the RBF kernel. The results of this model were very near to random guess with accuracy of just near 55%. The reason of such poor results is because of all the training dataset was sampled at different frequencies thus pitch calculation was not precise. Moreover even after applying different filters, the speech samples still had noise in it which was added to pitch calculation. This approach was not satisfactory. The next sections show how I applied the knowledge I learnt from different research papers, books and the internet to build a real gender identification system using different MATLAB packages.

### 6.2.3 Algorithm

The algorithm of the final gender identification system (SDC fused model) is completely different from the initial approach. As shown in chapter 4, most commonly used feature set for gender identification is MFCC. In [HC] has shown that long term features perform better than short term features. The idea behind long term features is that they are robust for noisy conditions and they capture most of the
information even if the speech signal has noise in it. In [HC] used first order statistics of MFCC to obtain a long term feature set. SDC are long term features that capture the majority of the information of the speech signal thus proved to perform better than first order statistics of MFCC. However using SDC alone did not give excellent results as shown later in Results section. [TYZ06], [DC11] [KH10] and [IKJGY10] have used fused systems to achieve more than 90% accuracy for gender identification. In these papers they have used MFCC along with pitch to determine the gender of the speaker. This shows that using MFCC with pitch increases the accuracy a lot. So I decided to use SDC along with pitch as SDC are long term features and pitch is the fundamental discriminating factor.

To get satisfactory results pre-processing by different speech enhancement methods was needed. The remaining section shows the pre-processing applied to training speech samples. First all the data was resampled to 44 KHz then to reduce the data and remove data which can disrupt the training model, a voice activity detector was used to remove non speech frames and silence. After that spectral subtraction was applied to the signal to remove background noise, white noise and musical noise.

SDC Extraction

As I used a fused model so two types of features were needed, SDC and pitch. This section shows the method I followed to extract SDC features. MFCC and SDC were extracted from the speech corpus that was obtained as the result of the pre-processing techniques applied to the dataset. To calculate SDC, MFCC had to be calculated first. MFCC was calculated using voicebox. 12 MFCC features were calculated for every frame 30 millisecond window with an overlap of 20 milliseconds. MFCC was then passed to mfc2sdc function by[Sah12]. The $N - d - p - k$ parameters are assigned as $7 - 1 - 3 - 7$. Finally the mean of the calculated SDC was normalized to zero and the standard deviation of 1. The block diagram of the SDC extraction step can be seen in the figure 6.1.
6.2. SYSTEM DESIGN

Figure 6.1: Block diagram of SDC feature extraction

Pitch Extraction

The pitch was extracted using the sub harmonic to harmonic ratio explained in section 4.1.3. The MATLAB function provided by [Sun02] was used to extract pitch. Frequencies lower than 100 Hz and higher than 300 Hz were ignored using filters. Every speech file was resampled to 44 KHz. The MATLAB function by [Sun02] calculated the pitch of every 25 millisecond window so the mean of all the windows was taken. The result was considered as the mean of the input file.

Model Training

As described in [TYZ06] gender identification is a general audio classification problem with two datasets: male speech dataset and female speech dataset. For training the model, data was prepared using the steps described in the previous section. For training the GMM models the algorithm defined in [TYZ06] was used. Given the $T$ parameterized SDC feature matrix for each speech segment $X$, the following expression was computed for each GMM to find the acoustic log likelihood scores.

$$S_g = \sum_{t=1}^{T} \log \left( \sum_{j=1}^{M} p_j b_j(x) \right) \quad (6.1)$$
Where $M$ is the number of Gaussian Mixture components and $b_j(x)$ are unimodal densities; $p_j$ are corresponding mixture weights. Using the method described I trained 3 GMM models using the SDC feature set. The number of components chosen was 8, 16 and 32. I tried to train the model with a higher number of components but that resulted in over fitting of data. A Score $S = S_{Male} - S_{Female}$ was calculated, if the $S$ score was higher than 0, the speech was labelled as male else female otherwise.

In order to identify the gender based on pitch, data was pre-processed as shown in the previous sections. The pitch for each frame was calculated using harmonic to sub harmonic pitch estimation method and the average of all the pitch was used as the key. Regardless of the non-linear separability of the data, I set the 205 Hz as the threshold frequency to distinguish the gender. Speech samples with an average pitch of less than 205 Hz were labelled as male else female otherwise. The reason of choosing this frequency threshold was to simplify the mathematical calculations that were required to make a fused model equation.

To fuse the two approaches, I scaled the both scores to the same dimension and did a weighted sum to get the final answer using the equation provided by [TYZ06] shown below.

$$S = \frac{(S_{sdc} - \lambda_{sdc})}{\alpha} + \frac{(\lambda_{sdc} - S_{pitch})}{(100 \times p)} \quad (6.2)$$

Where $S_{sdc}$ is the difference of the score of the male model log likelihood and female model log likelihood. $S_{pitch}$ is the mean pitch of the speech utterance. $\lambda_{sdc}$ and $\lambda_{pitch}$ are the thresholds which are described above. $\alpha$ and $p$ is the parameter that adjusts the weight given to each approach. For the fused model I assigned the value of 0.6 to $\alpha$ and 0.4 to $p$. According to equation 6.2, the gender was assigned to the speech segment based on the $S$ score. If the score was above 0, the speech was labelled as male speech else female otherwise.

The block diagram of the system can be seen in figure 6.2. The SVM model trained for the fused model was trained on a different dataset from the initial approach in which resampling was not performed which caused errors. Moreover that pitch model was trained using a non-linear SVM.
Dataset

There are many speech corpora available on the internet like TIMIT but most of them were not free and was out of my budget to purchase so for this purpose I made a dataset by collecting speech from many different sources and speech corpora. The male dataset consisted of speech utterances from Voxforge speech corpus and Quran recordings from “Quran Mp3 with Urdu Translation” android app\(^1\). The female dataset is consisted of speech utterances from Voxforge and Shruti Bengali\(^2\) speech corpora.

The dataset was formed by 10 speakers from each gender for total of about 1 hour of speech per gender which almost 6 minutes for every speaker. The complete training set consists of 12 languages to make sure that the system is independent from language and dialect of the speaker. The languages that were included in the dataset are English, Urdu, Arabic, Italian, Spanish, French, German and Indian regional languages like Bengali, Telugu, Tamil, Marathi and Gujarati. The English language was kept as a major language because English speech utterances were available in abundance.

---
\(^1\)Voxforge: [http://voxforge.org/home/listen](http://voxforge.org/home/listen)
\(^2\)Quran Mp3 with Urdu Translation: [https://play.google.com/store/apps/details?id=mydotdev.quranurdu2&hl=en_GB]
\(^2\)Shruti Bengali: [http://nltr.org/snltr-software/resources.php](http://nltr.org/snltr-software/resources.php)
The speeches were recorded at different frequencies ranging from 8 KHz to 48 KHz. All speech utterances were resampled to 44 KHz to reduce any channel noise. One hour of speech seems little as many hours of speech has been used in research papers but I made a compromise as I had to perform a different number of experiments. Indeed with this much data, training one 1024 component GMM took 3 days but in the end the data started to over fit with large number of GMM components.

### 6.2.4 Feature Selection

To find that whether SDC features perform better than MFCC features, I developed a test in which I trained GMM models with 8, 16 and 32 components with both MFCC and SDC features. After experiments I found out that SDC feature does perform better than MFCC feature. The features used for training the GMM was 12 coefficients of MFCC and $7 - 1 - 3 - 7$ coefficients of SDC which makes an 84 dimensional vector.

**Convergence**

As I intended to perform a really large number of tests, I skipped the convergence test for GMM in which we can determine the optimal number iterations needed for the GMM to converge. For this purpose, I used the default parameters of *Netlab* by setting `option(8)` true. This option decides the number of iterations required for convergence of data. The optimal number of iterations that were select was 20.

**Graphical User Interface**

A Graphical User Interface has been developed to test the system in real time using the recording feature through a microphone. First the users need to set which type of model they want to use using the Radio buttons. Then they can select the number of GMM components using the drop down list. After that user can record their speech by using the record button. As the recording finishes, a plot of the waveform and the recording’s spectrogram appears. Speech enhancement and feature extraction as
6.2. SYSTEM DESIGN

described in algorithms are implemented and then the result is calculated depending upon the model user has used. Finally a pop window appears that shows the gender of the speaker. The graphical user interface is made a bit complex that has the ability to select different models if any user wants to see the result of different models.

Figure 6.3: Snapshot of the Graphical User Interface of the system
6.3 Experiments and Results

In chapter 4, I showed that there are three different types of method that are used to develop gender identification system. In this section I will evaluate the performance of these 3 different types of models. Results obtained from testing on the same speakers on which the models were trained are ignored as in most cases the accuracy was 100%. The results given below are from close set identification.

The motivation behind performing these experiments was to determine the performance of each type of model trained on same amount of speech so that a reference of performance can be established. In literature, many papers can be found that have used MFCC with GMM to identify the gender but there is no paper exists that uses SDC for gender identification. To compare the performance of MFCC and SDC, I trained models with exactly same amount of data and parameters so that a baseline can be established so that a difference in performance can be analyzed between these two types of models.

6.3.1 Pitch Based Models

Pitch based models are those models which use only pitch as discriminating factor to identify the gender of the speaker. For training the model, the data was prepared by applying the pre-processing explained earlier. After that pitch for every frame was estimated using the harmonic to sub harmonic method and the average of pitch was used as a key. Using the training data I trained non-linear SVM using the RBF kernel to identify the gender of the speech. For testing, the pitch was estimated using the same method and the mean of the pitch was passed to the model as an input for classification.

I performed different types of experiments to evaluate the performance of the pitch based models. The primary motivation behind these experiments was to examine the factor of speaker variability in pitch based gender identification model. The secondary motivation behind performing these experiments was to determine

---

3I have done a detailed research about finding a paper that uses SDC to find gender but I could not find any. My claim is based on my search. There might exist such paper but I was not able to find it.
the training settings at which pitch based models perform highest and to understand
the behaviour of pitch based models when trained with speeches from different lan-
guages in different conditions. The length of each speech file is between 1.5 second
and 4 seconds.

**One Male Speaker vs. One Female Speaker**

For this experiment I trained a nonlinear SVM with RBF kernel. The training set
consisted of utterances from one female and one male speaker. The total time of
training speech was 6 minutes from each gender. The model was tested on the
utterances of different speakers. During each iteration a new speech file was used
as an input to the model. The results can be seen in the table 6.1 The length of the

<table>
<thead>
<tr>
<th></th>
<th>Male</th>
<th>Female</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Results</td>
<td>73.9568%</td>
<td>85.93%</td>
<td>82.0632%</td>
</tr>
</tbody>
</table>

Table 6.1: Results from pitch based model trained with 1 male and 1 female speaker

speech provided to model from each gender was almost same but the numbers of
files provided to the system were different. That is the reason that “Total” accuracy
of the system is not the average of the Male and Female results

**Multiple Male Speakers vs. One Female Speaker**

For this experiment I trained a nonlinear SVM with RBF kernel. The training set
consisted of utterances from one female (trained and tested with all ten female
speakers in the dataset ) and nine male speakers. The model was tested on the
utterances of different speakers. During each iteration a new speech file was used
as an input to the model. The results can be seen in the table 6.2 The length of the
speech provided to model from each gender was almost same but the numbers of
files provided to the system were different. That is the reason that “Total” accuracy
of the system is not the average of the Male and Female results
### Chapter 6. System Design and Implementation

#### Table 6.2: Results from pitch based model trained with 9 male and 1 female speakers

<table>
<thead>
<tr>
<th>Female Speaker No</th>
<th>Male</th>
<th>Female</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>99.5683%</td>
<td>45.5045%</td>
<td>62.9647%</td>
</tr>
<tr>
<td>2</td>
<td>97.4101%</td>
<td>45.4358%</td>
<td>62.2212%</td>
</tr>
<tr>
<td>3</td>
<td>98.7056%</td>
<td>49.4166%</td>
<td>65.3346%</td>
</tr>
<tr>
<td>4</td>
<td>98.4173%</td>
<td>46.2594%</td>
<td>63.1041%</td>
</tr>
<tr>
<td>5</td>
<td>99.7122%</td>
<td>47.083%</td>
<td>64.0799%</td>
</tr>
<tr>
<td>6</td>
<td>99.2806%</td>
<td>59.0254%</td>
<td>72.026%</td>
</tr>
<tr>
<td>7</td>
<td>99.2806%</td>
<td>52.4365%</td>
<td>67.5651%</td>
</tr>
<tr>
<td>8</td>
<td>99.2806%</td>
<td>50.7893%</td>
<td>64.8234%</td>
</tr>
<tr>
<td>9</td>
<td>99.2806%</td>
<td>55.8682%</td>
<td>68.4944%</td>
</tr>
<tr>
<td>10</td>
<td>99.2806%</td>
<td>49.1421%</td>
<td>63.8941%</td>
</tr>
</tbody>
</table>

#### One Male Speaker vs. Multiple Female Speakers

For this experiment I trained a nonlinear SVM with RBF kernel. The training set consisted of utterances from nine female and 1 male speakers (trained and tested with all ten male speakers in the dataset). The model was tested on the utterances of different speakers. During each iteration a new speech file was used as an input to the model. The results can be seen in the table 6.3. The length of the speech provided to model from each gender was almost same but the numbers of files

#### Table 6.3: Results from pitch based model trained with 1 male and 9 female speakers

<table>
<thead>
<tr>
<th>Male Speaker No</th>
<th>Male</th>
<th>Female</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>31.7986%</td>
<td>99.7941%</td>
<td>77.7881%</td>
</tr>
<tr>
<td>2</td>
<td>53.3813%</td>
<td>99.7255%</td>
<td>84.7584%</td>
</tr>
<tr>
<td>3</td>
<td>17.6978%</td>
<td>99.7941%</td>
<td>73.2807%</td>
</tr>
<tr>
<td>4</td>
<td>22.5899%</td>
<td>99.8627%</td>
<td>74.9071%</td>
</tr>
<tr>
<td>5</td>
<td>40.4317%</td>
<td>99.7255%</td>
<td>80.5762%</td>
</tr>
<tr>
<td>6</td>
<td>33.3813%</td>
<td>99.7941%</td>
<td>78.3457%</td>
</tr>
<tr>
<td>7</td>
<td>43.741%</td>
<td>100%</td>
<td>81.8309%</td>
</tr>
<tr>
<td>8</td>
<td>54.1007%</td>
<td>99.7941%</td>
<td>85.0372%</td>
</tr>
<tr>
<td>9</td>
<td>48.6331%</td>
<td>99.7941%</td>
<td>83.2714%</td>
</tr>
<tr>
<td>10</td>
<td>45.8421%</td>
<td>100%</td>
<td>68.1691%</td>
</tr>
</tbody>
</table>
provided to the system were different. That is the reason that “Total” accuracy of the system is not the average of the Male and Female results.

**Multiple Male Speakers vs. Multiple Female Speakers**

For this experiment I trained a nonlinear SVM with RBF kernel. As the total dataset contains ten male and ten female speakers, I divided the dataset in 5 folds in which every fold has 2 speakers from each gender. The models were trained on eight speakers from each gender and trained on the remaining two speakers from each gender thus creating a 5 x 2 cross validation test. During each iteration a new speech file was used as an input to the model. The results can be seen in the table 6.4. The length of the speech provided to model from each gender was almost same but the numbers of files provided to the system were different. That is the reason that “Total” accuracy of the system is not the average of the Male and Female results.

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>75.6835%</td>
<td>96.4997%</td>
<td>88.1395%</td>
</tr>
<tr>
<td>2</td>
<td>79.1367%</td>
<td>95.5388%</td>
<td>89.7674%</td>
</tr>
<tr>
<td>3</td>
<td>77.6978%</td>
<td>96.2251%</td>
<td>88.8372%</td>
</tr>
<tr>
<td>4</td>
<td>77.4101%</td>
<td>96.2251%</td>
<td>92.3256%</td>
</tr>
<tr>
<td>5</td>
<td>78.705%</td>
<td>95.6074%</td>
<td>89.1204%</td>
</tr>
</tbody>
</table>

Table 6.4: Results from pitch based model trained with 8 male and 8 female speakers.

As we can see from the results shown above that in starting three experiments, the average “Total” accuracy is about 70%. The experiments in which one gender speech was dominant performed extremely well for that gender but results for the other gender were not promising. Theoretically, the performance should not be dependent on the amount of data for each gender as we know that there exists a threshold pitch value that discriminates male speech from female speech but in this case that did not happen. The reason is that different additive and channel noises corrupt the speech signal. Despite of applying all those speech enhancement techniques on the signal, these noises still degrade the performance.
In the last experiment we can an average “Total” accuracy of almost 90% which is a good performance. The reason behind these good results is that SVM was able to make a clear non-linear separating hyper plane because in this experiment, it had enough data to make a clear separating hyper plane. These experiments covered the factor of speaker variability as to cover a large amount of speaker; we need large amount of training data as well for covering different speakers.

6.3.2 Models Based on Acoustic Features

Acoustic models are those models in which acoustic features like SDC and MFCC are used to develop a gender identification system. In this section I will evaluate the performance of the acoustic models. I trained models using both MFCC and SDC features using different number of components in GMM.

MFCC Based Model

For training MFCC based model, the speech data was resampled to 44 KHz, pre-processing was applied on the data to remove any noise and silence. After that the data was filtered using 12 Mel spaced filter banks. The logarithmic energy output was passed for DCT to obtain cepstral coefficients. Then 12 MFCC features were used to train a GMM for each gender.

The number of components chosen was 8, 16 and 32 for each model. Two models (one male model, one female model) were trained for every training dataset for each certain number of GMM components as shown in table 6.5. That means ten models (five male models, five female models) were trained with 8 GMM components. Similar process was repeated for 16 and 32 GMM components. In total 30 GMM models were trained, fifteen for each gender. Each training dataset’s speech time was almost 48 minutes for each gender consisting of 8 speakers. For testing MFCC for speech file were passed to each GMM and log-likelihood for each gender was calculated.

Finally the score of female model was subtracted from score of a male model. If the score was higher than 0, then the speech was labelled as male else female
otherwise. This score calculation can be mathematically written as

$$ S = S_{Male} - S_{Female} $$  \hspace{1cm} (6.3)

Where $S_{Male}$ is male GMM trained using MFCC and $S_{Female}$ is the female GMM. The results can be seen in the table 6.5, 6.6 and 6.7

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>62.0689%</td>
<td>89.7213%</td>
</tr>
<tr>
<td>2</td>
<td>66%</td>
<td>93.0348%</td>
</tr>
<tr>
<td>3</td>
<td>85.9649%</td>
<td>94.2857%</td>
</tr>
<tr>
<td>4</td>
<td>58.5526%</td>
<td>92%</td>
</tr>
<tr>
<td>5</td>
<td>56.4731%</td>
<td>96.3503%</td>
</tr>
</tbody>
</table>

Table 6.5: Results from MFCC model trained using 8 GMM components

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>66.37931%</td>
<td>91.8613%</td>
</tr>
<tr>
<td>2</td>
<td>68.6248%</td>
<td>91.5%</td>
</tr>
<tr>
<td>3</td>
<td>87.7192%</td>
<td>100%</td>
</tr>
<tr>
<td>4</td>
<td>60.5263%</td>
<td>90%</td>
</tr>
<tr>
<td>5</td>
<td>59.8913%</td>
<td>97.3236%</td>
</tr>
</tbody>
</table>

Table 6.6: Results from MFCC model trained using 16 GMM components

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>89.6551%</td>
<td>99.3355%</td>
</tr>
<tr>
<td>2</td>
<td>76%</td>
<td>97.5%</td>
</tr>
<tr>
<td>3</td>
<td>92.9824%</td>
<td>100%</td>
</tr>
<tr>
<td>4</td>
<td>70.6667%</td>
<td>91.3333%</td>
</tr>
<tr>
<td>5</td>
<td>62.8089%</td>
<td>97.8102%</td>
</tr>
</tbody>
</table>

Table 6.7: Results from MFCC model trained using 32 GMM components
SDC Based Model

For training SDC based model, the speech data was resampled to 44 KHz and preprocessed like shown in the previous section. An 84 dimensional SDC feature set was used to train different models.

The number of components chosen was 8, 16 and 32 for each model. Two models (one male model, one female model) were trained for every training dataset for each certain number of GMM components as shown in table 6.8. That means ten models (five male models, five female models) were trained with 8 GMM components. Similar process was repeated for 16 and 32 GMM components. In total 30 GMM models were trained, fifteen for each gender. Each training dataset’s speech time was almost 48 minutes for each gender consisting of 8 speakers. For testing, SDC for speech file were passed to each GMM and log-likelihood for each gender was calculated.

Finally the score of female model was subtracted from score of a male model. If the score was higher than 0, then the speech was labelled as male else female otherwise. This score calculation can be mathematically written as

\[ S = S_{Male} - S_{Female} \] (6.4)

Where \( S_{Male} \) is male GMM trained using SDC and \( S_{Female} \) is the female GMM. The results can be seen in the table 6.8, 6.9 and 6.10

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>71.5517%</td>
<td>90%</td>
</tr>
<tr>
<td>2</td>
<td>71%</td>
<td>94.521%</td>
</tr>
<tr>
<td>3</td>
<td>86.4912%</td>
<td>93.8775%</td>
</tr>
<tr>
<td>4</td>
<td>71.0526%</td>
<td>91.6666%</td>
</tr>
<tr>
<td>5</td>
<td>64.0449%</td>
<td>93.7737%</td>
</tr>
</tbody>
</table>

Table 6.8: Results from SDC model trained using 8 GMM components

By comparing the results of MFCC model with SDC model, we can see that SDC performs better than MFCC models. Both models were trained on same type and same amount of data thus we can make a reference about performance of each
6.3. EXPERIMENTS AND RESULTS

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>86.2068%</td>
<td>94.6666%</td>
</tr>
<tr>
<td>2</td>
<td>90%</td>
<td>88%</td>
</tr>
<tr>
<td>3</td>
<td>98.2456%</td>
<td>99.59%</td>
</tr>
<tr>
<td>4</td>
<td>77.6315%</td>
<td>96%</td>
</tr>
<tr>
<td>5</td>
<td>77.5280%</td>
<td>97.3236%</td>
</tr>
</tbody>
</table>

Table 6.9: Results from SDC model trained using 16 GMM components

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>94.8275%</td>
<td>93.0232%</td>
</tr>
<tr>
<td>2</td>
<td>92%</td>
<td>94.5%</td>
</tr>
<tr>
<td>3</td>
<td>94.7368%</td>
<td>99.5918%</td>
</tr>
<tr>
<td>4</td>
<td>86.84210%</td>
<td>96.3333%</td>
</tr>
<tr>
<td>5</td>
<td>89.8876%</td>
<td>98.8321%</td>
</tr>
</tbody>
</table>

Table 6.10: Results from SDC model trained using 32 GMM components

model with respect to the other model. On average, the SDC model performed at least 10% better than the MFCC model regardless of the number of GMM components. The reason of this performance improvement is the usage of SDC features that provide robustness in noisy conditions.

The number of components of GMM also played an important role in determining the performance of the models. From results we can see that as I increased the number of the components of the models, the accuracy increased. The reason is that increasing the number of components can represent the data more accurately. But increasing the number of components can result into over fitting of data. I trained models with up to 1024 components (it took 3 days to train one SDC model with 1024 components) but data started to over fit.

6.3.3 Fused Model

In this section I will explain the results of my final model where I combined SDC with pitch to identify the gender of the speaker. The training of models is explained in model training section. The results can be seen in the table 6.11, 6.12 and 6.13. Generally tests are performed on speech utterances of different lengths like 3, 10
CHAPTER 6. SYSTEM DESIGN AND IMPLEMENTATION

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>94.0170%</td>
<td>99.3127%</td>
</tr>
<tr>
<td>2</td>
<td>88.9564%</td>
<td>95.7146%</td>
</tr>
<tr>
<td>3</td>
<td>94.7368%</td>
<td>99.5515%</td>
</tr>
<tr>
<td>4</td>
<td>98.0263%</td>
<td>98.1132%</td>
</tr>
<tr>
<td>5</td>
<td>98.8764%</td>
<td>97.0731%</td>
</tr>
</tbody>
</table>

Table 6.11: Results from fused model trained using 8 GMM components on SDC features

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>94.8717%</td>
<td>98.0066%</td>
</tr>
<tr>
<td>2</td>
<td>91.6784%</td>
<td>100%</td>
</tr>
<tr>
<td>3</td>
<td>85.6804%</td>
<td>99.5918%</td>
</tr>
<tr>
<td>4</td>
<td>98.6842%</td>
<td>98%</td>
</tr>
<tr>
<td>5</td>
<td>97.75280%</td>
<td>97.3236%</td>
</tr>
</tbody>
</table>

Table 6.12: Results from fused model trained using 16 GMM components on SDC features

and 30 seconds which I was not able to perform because during the pre-processing of my dataset, I modified my dataset in a way that I had speech files of varying length between 1.5 seconds to 4 seconds.

Open Set Identification

In this experiment I trained models with all the close set identification dataset (10 male, 10 female) and tested on a large number of speech files uttered by a very large group of speakers. The open set identification dataset has utterances from 89 male speakers and 22 female speakers. The difference in the number of female and male speakers is because getting female speech corpus was quite a difficult task and male speech was abundantly available on Voxforge. The length of the speech utterances are between 3 to 7 seconds. The results of these tests can be seen in table 6.14. As we can see that the miss percentage is higher in the pitch based and MFCC models. The reason is that though the pitch is the fundamental discriminating factor between male and female speech, it is not a good feature for identifying the gender because of the environmental noises one can find in real world speech. MFCC does extract
Table 6.13: Results from fused model trained using 32 GMM components on SDC features

<table>
<thead>
<tr>
<th>Training Dataset</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.5517%</td>
<td>98.3389%</td>
</tr>
<tr>
<td>2</td>
<td>96%</td>
<td>100%</td>
</tr>
<tr>
<td>3</td>
<td>94.7368%</td>
<td>100%</td>
</tr>
<tr>
<td>4</td>
<td>98.6842%</td>
<td>98.6667%</td>
</tr>
<tr>
<td>5</td>
<td>98.8764%</td>
<td>97.5669%</td>
</tr>
</tbody>
</table>

Table 6.14: Results from acoustic and fused models tested on large amount of data

<table>
<thead>
<tr>
<th>Model</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC with 8 GMM Components</td>
<td>57.3705%</td>
<td>72.0224%</td>
</tr>
<tr>
<td>MFCC with 16 GMM Components</td>
<td>59.31%</td>
<td>74.1460%</td>
</tr>
<tr>
<td>MFCC with 32 GMM Components</td>
<td>65%</td>
<td>74.6468%</td>
</tr>
<tr>
<td>SDC with 8 GMM Components</td>
<td>69.7742%</td>
<td>73.2584%</td>
</tr>
<tr>
<td>SDC with 16 GMM Components</td>
<td>72.0318%</td>
<td>74.6067%</td>
</tr>
<tr>
<td>SDC with 32 GMM Components</td>
<td>74.2974%</td>
<td>77.4157%</td>
</tr>
<tr>
<td>SDC Fused Model with 8 GMM Components</td>
<td>86.8073%</td>
<td>70.1011%</td>
</tr>
<tr>
<td>SDC Fused Model with 16 GMM Components</td>
<td>79.2828%</td>
<td>78.0898%</td>
</tr>
<tr>
<td>SDC Fused Model with 32 GMM Components</td>
<td>79.2828%</td>
<td>80.33707%</td>
</tr>
</tbody>
</table>

more information about speech but that did not perform well either especially for male speeches. The reason is that the male speech set was based on real world speech utterances that have a significant amount of noise in it. As MFCC is a short term feature and is not considered robust for noisy conditions thus its poor performance for male speech can be explained.

As expected the SDC based models has lower miss percentage than MFCC and pitch based models. The reason is the robustness SDC feature provides for noisy conditions. As we can see from the results that combining SDC and pitch greatly enhanced the performance of the model. The combination of noise robustness with basic discriminating factor performed better than all other models. The fused model achieved an accuracy of about 95% for male speech where other models gave an accuracy of almost 85%.

The number of Gaussian Mixture models also performed a significant amount
of role in the performances of the models. As we can see that as the number of components increased, the performance of the model increased. I tried to train models with higher number of models but the data started to over fit after 64 components.

6.3.4 YouTube Videos

I performed a simulation to evaluate the performance of the system if it was used to identify the gender of the speakers of the television programs. I took 5 speech segments from YouTube videos for each gender and used them to simulate real world conditions. I selected some video interviews, commencement speeches and TV shows to cover the problems like applause, cheer, background music and noise. The list of the YouTube videos can be seen in the appendix. The speech was re-

<table>
<thead>
<tr>
<th>Model</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC with 8 GMM Components</td>
<td>19.4805%</td>
<td>11.1111%</td>
</tr>
<tr>
<td>MFCC with 16 GMM Components</td>
<td>31.168%</td>
<td>29.6296%</td>
</tr>
<tr>
<td>MFCC with 32 GMM Components</td>
<td>36.3636%</td>
<td>37.0370%</td>
</tr>
<tr>
<td>SDC with 8 GMM Components</td>
<td>25.97402%</td>
<td>14.8148%</td>
</tr>
<tr>
<td>SDC with 16 GMM Components</td>
<td>58.4415%</td>
<td>44.4444%</td>
</tr>
<tr>
<td>SDC with 32 GMM Components</td>
<td>68.8311%</td>
<td>48.148%</td>
</tr>
<tr>
<td>SDC Fused Model with 8 GMM Components</td>
<td>84.4155%</td>
<td>81.481%</td>
</tr>
<tr>
<td>SDC Fused Model with 16 GMM Components</td>
<td>85.7142%</td>
<td>88.8889%</td>
</tr>
<tr>
<td>SDC Fused Model with 32 GMM Components</td>
<td>85.7142%</td>
<td>96.2962%</td>
</tr>
</tbody>
</table>

Table 6.15: Accuracy of all the models that were tested on YouTube Videos

sampled to 44 KHz to match the modelling samples. After that VAD and spectral subtraction was applied to remove any silence and background noises. After applying all the pre-processing, relevant features were extracted to test on the relevant model accordingly.

The results of this simulation can be seen in table 6.15 that shows the accuracy for 10 YouTube videos. As the table shows, the accuracy of fused model is exceptional as compared to MFCC and SDC models. The MFCC model performed worse on this test because for real time environments, capturing long term information of the signal is necessary to perform better in noisy environments in that MFCC fails
6.3. EXPERIMENTS AND RESULTS
to do so. SDC model performed quite better than SDC as SDC capture the majority information of the speech signal. In fact, the performance was highest in the speech files where there was less noise and high speech density like the “Yahoo CEO Interview” on which I tested my models on.
Chapter 7

Conclusion

7.1 Summary

This dissertation was about creating a gender identification system using speech. It showed all the necessary technical background and the steps required to make such a system like the digital form of speech, speech enhancement methods, classification method, and relevant features.

For the purpose of this project all three types of gender identification systems were built and their performance was evaluated on different tests using different number of parameters. Because of limited time, data and resources I could not perform training of large amounts of data. So work was reduced to simple models.

All three implementations were tested on different datasets having different languages, dialects and loudness. The SDC fusion method performed better than all other methods giving a result of almost of 80% accuracy for each gender when tested on a large number of speakers. The final implementation can also identify the gender of the speaker through a microphone because of the feature implemented in the GUI. Results performed on YouTube videos can be considered good considering the amount of training data used to train the models.
7.2 Future Recommendation

The system gave almost 80% accuracy on a large number of speakers this can be increased by adding more training data of different nature. As the data was recorded from different microphone with different conditions that caused some errors, steps can be performed to reduce this channel noise. Moreover if SVM used for pitch model can be trained using the Adaboost technique than performance can be increased a lot. Though using Adaboost on strong classifiers like SVM usually decreases the accuracy but as a pitch is only one dimensional data, accuracy will increase in this case. Tests should be performed using different lengths of speech utterances to find better results on different lengths of speech. A feature in which the model can identify the different number of male and female speakers speaking at the same time can be added. A UBM-GMM model can be added so that the accuracy across many different speakers can be increased. As I selected a frequency threshold for pitch classification in my fused model, I think work should be done in combining a non-linear SVM model with a SDC model to make this model perform better.
Bibliography


[BO00] Ronald J. Baken and Robert F. Orlikoff. *Clinical Measurement of*


[SF95] Robert Schapire and Yoav Freund. A decision-theoretic generalization of on-line learning and an application to boosting. Proceedings of


Appendix A

Appendix

File Listing

- Data

- Models
  - SDC
    * F_8.mat
      - F_16.mat
      - F_32.mat
      - F_64.mat
      - F_128.mat
      - F_256.mat
      - M_8.mat
      - M_16.mat
      - M_32.mat
      - M_64.mat
      - M_128.mat
      - M_256.mat
• Pitch
  • Pitch.mat (pitch model trained with all speakers)
• MFCC
  • MFCC_model.mat (contains MFCC models for both genders)
  – Extracted Features
    • MFCC
      • Female.mat (MFCC features for each female speaker)
      • Male.mat (MFCC feature for each male speaker)
    • SDC
      • SDC_F.mat (SDC features for each female speaker)
      • SDC_M.mat (SDC features for each male speaker)
    • Pitch
      • Female.mat
      • M_S_1.mat
      • M_S_2.mat
      • M_S_3.mat
      • M_S_4.mat
      • M_S_5.mat
      • M_S_6.mat
      • M_S_7.mat
      • M_S_8.mat
      • M_S_9.mat
      • M_S_10.mat

• MFCCExtraction
  – ExtractMFCC.m (extract MFCC features from Audio Files)

• PitchExtraction
APPENDIX A. APPENDIX

- Pitch.m
- shrp.m

- **Pre-Processing**
  - detectVoiced.m
  - ExtractSegments.m
  - findMaxima.m
  - ResampleSound.m
  - ShortTimeEnergy.m
  - SpectralCentroid.m

- **SDC Extraction**
  - mfcc2delta.m
  - mfcc2sdc.m

- **Testing**
  - GenderDetection.m
  - Testing.m

- **Toolbox**
  - libsvm-3.17
  - netlab3_3
  - voicebox

- **Training GMM**
  - TrainGMM.m
Female YouTube videos

http://www.youtube.com/watch?v=TLbUqKr1Zuk
http://www.youtube.com/watch?v=BjYU7Q7hqEI
http://www.youtube.com/watch?v=Bpd3rajj8xww
http://www.youtube.com/watch?v=wHGqp81z36c
http://www.youtube.com/watch?v=GMWFieBGR7c

Male YouTube videos

http://www.youtube.com/watch?v=_jPaYnaKVDk
http://www.youtube.com/watch?v=k6U-i4gXkLM
http://www.youtube.com/watch?v=HdZfkXLR27g
http://www.youtube.com/watch?v=UF8uR626Klc
http://www.youtube.com/watch?v=rie-hPVJ7Sw