ŠIAULIŲ UNIVERSITY

DEPARTMENT OF ENGINEERING

Anupama Sira Jagadeeshchandra

SPEAKER IDENTIFICATION AND VOICE IMPAIRMENTS DETECTION

Master Thesis

Supervisor:

Prof. Gintautas Daunys

Šiaulių, 2018

ACKNOWLEDGEMENT

I take this opportunity to express my gratitude to Šiauliai University for the support.

I considered this fortuity to thank Prof. Gintautas Daunys (supervisor) for abundant assistance, for guidance through the path of the research work and for farthest support.

I'm also grateful for the Department of Smart Engineering Production for all the given basics and theoretical knowledge and technical aspects through courses, and for all the sophisticated literacy by the degree.

And I count this moment to please my parents for all the moral support.

ABSTRACT

Speech Processing being one of the best biometric identification method involves many applications with advantages which facilitated for speaker identification approach. By recognizing those speech features, this research work has two different topics. First part is Text and Language Independent Speaker Identification with comparison of two types of identification classifiers. Features are extracted by extraction methods like Mel Frequency Cepstral Co-efficient, I-Vectors and Vector Quantization and further classified by Euclidean Distance and Probabilistic Linear Discriminant Analysis classifiers for the two different methodologies. The comparison is between the Euclidean Distance and PLDA classifiers for the different performance measures such as Equal Error Rate, Thresholding (False Rejection Rate and False Acceptance Rate), Decision Error Trade-off, Receiver Operating Characteristics and Detection Cost Function. These performance measures are considered according to the NIST 2016 Speaker Recognition Evaluation Plan. In further, to prove that these methodologies perform for different languages, the databases like Libri Speech (English), Uyghur Language (Chinese) and Korean Languages are considered with both male and female speakers of large utterances. As the algorithm works for the classification for Euclidean Distance depend on the thresholding but for Probabilistic LDA depends on the LDA for reduced dimensionality which offer to check for different dimensions and hence results in best output. Thus, the main goal of this implementation is to get lower Equal Error Rate and Detection Cost Function to ensure that the methodologies give maximum accuracy and precision in text-independent and in multilingual conditions.

As the second part, using the same techniques like Mel Frequency Cepstral Co-efficient, I-Vectors for features extraction and Support Vector machines for features classification for Voice Impairments Detection is undertaken. The TORGO dysarthric speech databases which contains 3 female and 5 male abnormal voices with different utterances for the classification and Libri Speech for the normal voice samples are considered. This methodology is the best for the classification unless support vector machine is also used for regression in wide range. In this case, SVM is considered for classification in latest version of the MATLAB i.e., MATLAB R2018a for the coding and implementation. As performance measures, accuracy, precision, recall, sensitivity and specificities are calculated as per the data classified by the methodology.

And in furtherance, implementation and algorithm flow are explained in detail with experimental results in the enclosure.

TABLE OF CONTENTS

ABBREVATIONS

MFCC	Mel Frequency Cepstral Co-Efficient
VQLBG	Vector Quantization (LBG Algorithm)
PLDA	Probabilistic Linear Discriminant Analysis
LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
EER	Equal Error Rate
DET	Detection error trade-off
DCF	Detection cost function
FAR	False Acceptance rate
FRR	False rejection rate
LBP	Linear Binary Patterns
GMM	Gaussian Mixture models
LPC	Linear predictive coding
PLP	Perceptual Linear Predictive coefficients
FFT	Fast-Fourier Transform
UBM	Universal Background models
JFA	Joint Factor Analysis
GA	Genetic algorithm
ROC	Receiver Operating Characteristics
LLR	Log likelihood ratio

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Biometrics authentication or realistic authentication became a leading identification and access control techniques in this modern day. The present of digital security is facing difficulty to safeguard the integrity of the system. If other identity verifications like keys, passwords, pin and card so forth are computerized, they might be forgotten/ often stolen / lost and lead to cease accessing into the protected system. However, biometrics gives complete solution for this sort of dilemmas by means of divers' technologies. A biometric enforced system can be in two steps: recognition and verification. Recognition defines as one to many, a comparison of a person registered for database with the entire system, and Verification defines as one to one, a comparison of a pair of individual to determine whether they are from the same person or not. These authorizations necessitate comparing a registered or pre-stored biometric sample against a newly inspecting sample. In basic, it is a three steps process, create database, process features and test the against.

A biometric system fundamentally a pattern recognition system, which identify an individual by an accurate physiological or behavioural characteristics demoniac by the user. Some of them are,

- Finger Prints
- Facial features
- Hand geometry
- Palm Print
- Eye/Iris patterns
- Signature and keystroke dynamics
- Speaker Recognition

Finger prints:

Finger prints biometric authentication is forsooth old ink and paper method to identify a particular person by specifically law enforcement agencies. A unique point in fingerprint is that even twins does not have the same print, probably one in a million might have, thus it will create a distinct authentication. Nowadays its applications are increasingly helpful in many areas such as large-scale Automated finger imaging systems which is used by law purposes, data stored computers, Smart phones and gadgets etc.,

Facial Features:

Facial recognition is an endoscopic process where the clients face is stored as photograph and the resulting image will be scanned for the output of the verification. But it will be difficult for the developers to maintain the stability if it reaches to the vast database and also to sustain performance in recognizing because of the identical or very similar face structures, one can fool the scanners.

Hand Geometry and Palm prints.

Hand or finger geometry is referred as an automated measurement of many dimensions of individual hand and fingers. Palm prints will be slightly changed, alike fingerprint method, it will take the complete scan of the palm. Since its size and accessing area is bigger, it is not so popular and limited to use in workstations or in smart technologies and devices.

Iris Pattern or Eye pattern:

Iris scanning or iris pattern scaling techniques measures the iris pattern in the coloured portion of the eye. Although it is not considered as biometric, eye and iris identification are still biometric recognition. Meanwhile iris patterns are created randomly, unlike fingerprints, this is also not same for not even for twins and further not even to their left and right eyes. It is used for both identification and verification, applications will work perfectly because of its number of degrees of freedom. But it also results in major disadvantage for the relatively memory-intensive storage problems.

Signature and Keystroke dynamics:

Signature authentication is also one of the oldest method as fingerprint, to identify a person by examining the signature and unlike keystroke is an updated version of signature, like with how much speed and pressure the client did it and the total time for typing something, or on the time gaps between the strokes, etc., it also gives high rate recognition for and individual but with the drawback that if its observed or seen flawlessly by imposter, it is possible to fool the computer easily, and thus this session is still under improvement [10].

1.1. Speaker Identification: Speaker identification, unlike other methods it stores the database and then verify with the tested ones. Thus, it can be explained in two steps, Speaker recognition and speaker verification.

Speaker Recognition: Automatic recognition of speaker includes speech waves, it has two sessions. First will be the one where enrolled and training sessions will be conducted and the second is referred as the testing sessions. In recognition, it acts as one to many where the given voice is recognized as client or imposter by the testing it against the database.

Speaker Verification: In this case, it is one to one matching unlike other methods, in which it will verify whether the testing speaker claims to be him or her is the same or not. Each registered speaker, since their voices are stored in database, it is possible to identify the exact client if not an imposter. These all of the explanation are demonstrated in the below diagrams.

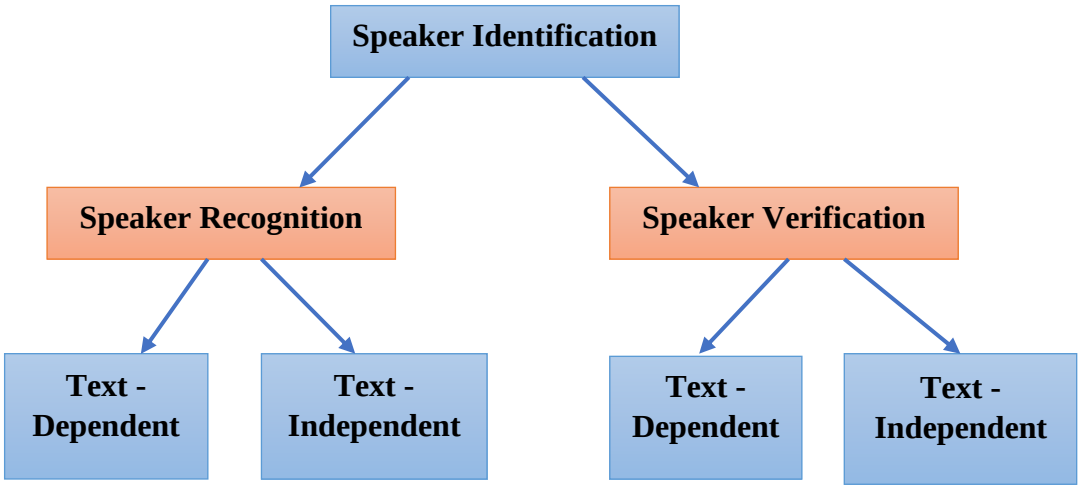


Figure 1: The scope of Speaker Identification

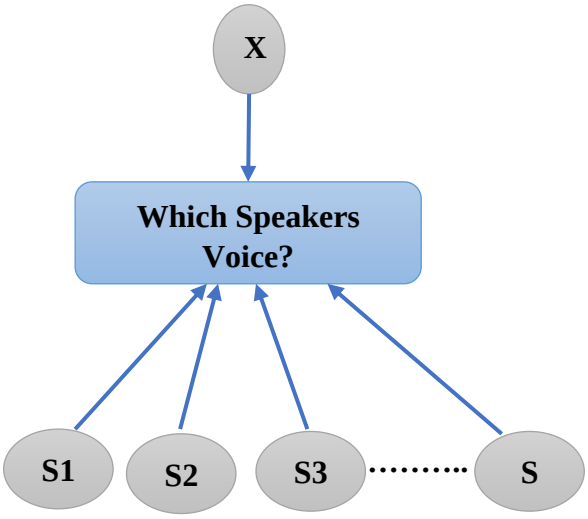


Figure 3: Basic Speaker Recognition

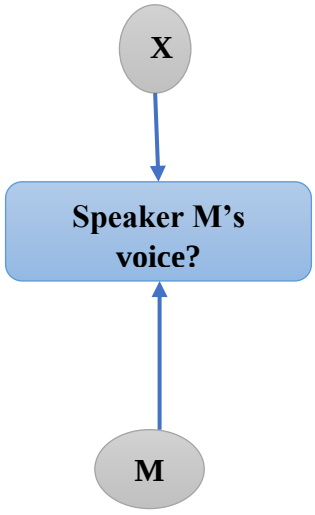


Figure 2 : Basic Speaker Verification

Further, it is important to note that speaker identification has different classifications like text-dependent and text-independent. (Figure 1).

One of the goal of this study is to identify the speaker, so the complete focus is on speaker recognition. Further, the above figures 2 &3 represent the speaker recognition and verification. The process involves 3 identification sub-processes: Feature extraction, Feature Classification and Feature matching. Feature extraction is the process of extraction of small data of a speaker from his/her voice signal to represent the person. Feature classification is the process where it reduces the dimensionality and big data of features specifically or accurately for the next process. And that next process is feature matching, which involves the actual procedure of the identification, where it tries to matches the features of unknown to the database, and within extracted features if something matches it returns the identity, if not it will be the imposter. The complete detailed explanation of all these processes are in the next chapter.

It is also important to note the application and advantages of this concept. Speaker identification is used by the organizations and for the purposes as follows [11],

- Government: For general operations, police, revenue departments, insurance systems etc.,
- Security and Surveillance: credit card transactions, monitoring etc.,
- Financial services: Associated Banks, Securities markets, government banks, visa and insurances, funds etc.,
- Forensic: Cyber security police, lawful intercepts, law enforcements, CID's Ministry of justice, prosecutors, general bodies and civil people etc.,
- Telecom and Call centre:
- Healthcare: Health management
- Transportation: Aeroplane, railways and interarea transportations security.
- Multi speaker tracking: Conferences, Meetings etc.,
- Personalized user interfaces: Voice-mail, telephone aided services, smartphones and gadgets etc.,
- Transaction authentication: Bank transactions, self-administration, password reset over phone, prison telephone usage etc.,
- Information retrieval: customer information for call centre, speech skimming.

- ❖ Access control: voice opened locks, computers and data networks, border control, and also for smart devices.
- ❖ Remote time and attendance logging: Employee time tracking. Home parole verification etc.,
- ❖ Audio mining: Automatic indexation of audio broadcast, forensics on intercepted calls etc.,

1.2. Detect Voice Impairments: The second context of the thesis is to detect voice disorders, by means of dysarthric speeches to the normal ones. To be in detail as follows:

"Voice" referred as the production of noise/sound of some quality, and "Speech" referred as the combination of meaningful words or intelligible words. Thus, continuous voice production is collectively called as speech. In this speech production, it is a complicated meld of the respiratory driving force, the supremely complex laryngeal muscular interactions and the placement of tones in the respiratory system. There are many physiological, behavioural and many aspects of human interactions which can affect the voice production. And also, in terms of disorder, it will affect the daily life/routine of a person when it is influenced. In this context, one of that kind of disorder is dysarthria. When the speech is characterized by slow, slurred and difficult to understand is referred as dysarthric speech. It is one the pathological disorder caused by the weakness of muscles which are used for speech production, or by paralysis or inability to coordinate the muscle of the mouth, or by neurological contractions. This occur as a developmental disability and as a sign of neuromuscular disorder like cerebral palsy and Parkinson's disease. [12]

The automatic detection of voice impairments in this case is distinctly a different modelling strategy. Using the speaker identification tools and methodologies, this research represents a unique and very accurate methodology to detect the dysarthric pathological voices from the healthy ones. Acoustic analysis is a useful MATLAB tool to diagnose such cases, with the advantages like it's a non-invasive and objective diagnosis complementary tool based on the observation of the vocal folds. The state-of-art in acoustic analysis leads us to identify and estimate a huge considerable term of acoustic parameters such as jitter, frequency, pitch, shimmer, harmonics to noise ratio, indexes, amplitudes and glottal noise ratios etc., thus by considering the database of such kind, and using few methodologies, it is possible to give 100% accuracy and in the next chapters, detailed explanation can be found with experimental results.

1.3. Outline of the Thesis: The intention of the introductory section is to furnish a general idea for considered tasks and this is the Chapter 1.

Chapter 2: Literature review, where few of the best previous works and analysis methods are mentioned and considered those as motivational works.

Chapter 3: Identification strategies, represent the detailed explanations of the overall methods used and also the definitions and the working procedures.

Chapter 4: Performance measures, in which all performed calculation definitions and explanations are portrayed.

Chapter 5: Complete Architectures and Specifications of overall implementations are in this section.

Chapter 6: Speaker identification and comparisons of methodologies and their experimental results are described with the explanations of Databases and real-time implementation with results.

Chapter 7: Voice impairments detection by identifying the abnormal signals over normal ones using Support vector machine algorithm to give very precise result.

Chapter 8: Conclusion, which gives a light summary about the whole concept and answers for the questions aroused in the experiments.

Chapter 9: References where all the papers, books, websites which are used are mentioned and cited and also the databases links which are utilized by the programs.

1.4. Aim and Objectives:

Aim:

i. Text and Language independent speaker identification with new methodologies.
ii. Voice impairments detection from pathological voice samples.

Objectives:

i. Create two new different methodologies for mentioned trials for meliorate gains.
ii. Comparison of performance measures for the developed approaches which are designed for speaker identification.
iii. Performance measures of Voice disorder detection using dysarthric speeches.
iv. An application, real-time integration and execution using the Euclidean Distance as feature matching with other techniques.

2. LITERATURE REVIEW

This review will be the explanation of previous different methodologies for various similar experiments to the proposed one. It also gives information about that past implementations, considered parameters and comparable performance measures.

2.1 Speaker Recognition Implementations and Results:

Automatic voice recognition algorithm which includes MFCC and VQ with 10speakers database to provide the error rate is 10% to 18% prove that feature extraction and classifier gives the better results. And also, to classify the speakers, calculation of distance is worthful [1]. The database which are created using different speakers recorded from microphone and telephone speech are also recognized and classified in i-vector method. In this collaboration of different methods like i-vector, Joint factor analysis, LDA, SVM are all used to give the best results and hence proved that with the limited small database, it is possible to obtain good results [2]. I-vector with different normalization types are explained with the others methods like PLDA and i-supervectors which also results in better performance measures like EER, DCF and DET curves. A detailed comparison of PLDA for few normalization values, i-vector with other improved algorithm are well explained and proved that the PLDA implementation are better for speaker recognition with both male and female speakers. Another important point to note here is text-dependent and text-independent methods, so this journal which also states that those methods are implemented for text-independent method [3]. Another study which is similar to this context is [9]where PLDA approach for language and text independent speaker recognition is experimented. In this State-of-the-art PLDA aims at modelling all the sources of undesirable variability within a single covariance matrix. Although the normalization techniques are improved, it is important to try with new implementations and in this it showed improved results up to l0 % EER and 6.4 % in minimum DCF. [Using the basic MFCC technique and VQLBG algorithm for extraction and to minimize the data of extracted features are studied for the purpose to include and verify the number of centroids and type of windows in MFCC. Thus, it is stated that number of increase in centroids will result in increase in identification rate of the system and also it has been found that the hamming windows with the combination of mel frequencies gives the best performances. Comparison between 1 to 64 number of centroids and triangular, rectangular and hamming window experiments were done and proved the above [4]. The weighted Euclidean Distance for speech recognition is stated and proved, such that all steps of the process are explained well and using MATLAB they

simulated an environment which shows the distance between the possibilities of speakers like imposters and original clients. By this way, it is assured to use the Euclidean Distance [5].

2.2 Voice Impairments Detection:

Automatic detection of voice disorder includes the detection of pathological voice activities by means of different techniques. One of that pathological disorder detection in which dysphonic voices will be detected is experimented by many researches. One study of that is the comparison of Linear binary patters(LBP) and Mel frequency coefficients. In this research, the comparison between LBP and MFCC are experimented for healthy and pathological voice samples to differentiate them from the other and it is stated that LBP gives the better performance in terms of Accuracy, Precision, Sensitivity and Specificity when compared to MFCC [6]. Other experiments which uses GMM and MFCC to build an automatic detection system which is capable of differentiating normal and pathological voices were conducted. A sustained vowel is easier than the detection with continuous speech with the detection rate up to 91.66%. Detection with continuous speech is more realistic compared to sustained vowel because sustainable vowel is not used in daily life [7]. And the final mentioning paper is very similar to the one which is implemented in this context, pathological voice analysis to detect neurological disorders using MFCC and SVM. In this work, it is provided a novel method which is relative fast and designed for this task. SVM classifiers basic principles are explained in detailed and the experiment is conducted for normal and diseased subjects ranged from 1.5 seconds to obtain 93% accuracy [8].

From the above survey consequently, it is proved that many experiments and simulation works are done based on many solely techniques and combined methodologies to give the best results for stated objectives.

3. IDENTIFICATION STRATEGIES

Information in a speech for speaker identification can be used in many ways for the recognizing systems. Although the systems have two models, recognition and verification, also includes this information for the process. Usage of voice for speech processing, works like this: recording of voice sample/ speech sample to register one's voice, which is referred as "Enrolment", and if different utterances are required, multiple recordings will be recorded which is referred as "Training" and the storage of samples is called as database. And finally, to verify the speaker, record again either the same person as client or different person as imposter and hence this will be termed as "Testing". The sample from testing, if it matches, speaker is recognized and if not, it is an imposter.

Thus, in this way modules steps will be in the flow. To finalize, Speaker identification will further be sorted as "Feature Extraction", "Feature Classification" and "Feature Matching". Typically, in speaker recognition there will be only two modules, feature extraction and feature matching. In few methodologies, one classification or selection process will be added, but to improve the efficiency of the algorithm and to testify the new methods, 2 layers of classification is included in this research work. Continuing for the second part, that is to detect for voice impairments also the same procedure will be followed nonetheless with 1 classifier and different identification methods. Eventually to list the identification methods for this context are:

- Feature Extraction
- Feature Classification/Selection
- Feature Matching

To interpret these techniques, few of the different methods will be explained in further.

3.1 Feature Extraction: Feature extraction is often referred as the signal processing frond-end. The speech waveform will be converted to a type of parametric representation for the analysis and processing. This technique is used to reduce the dimension of input vector despite of maintaining the perceptive power of the signal. The output features have the characteristics of the useful information of speech. Those kind of extraction techniques are Principal component analysis, Linear discriminate analysis, Mel-frequency cepstral co-efficient, Independent component analysis, Cepstral analysis, Mel-frequency scale analysis, Wavelet technique, Filter bank analysis, Linear predictive coding, Linear

prediction cepstral coefficients, Kernel based feature extraction method, Dynamic feature extractions, RASTA filtering, Spectral subtraction, Integrated phoneme subspace method and Cepstral mean subtraction. In these vast extractions, only best presented algorithms are explained further and for this research, one of them is considered.

3.1.1 Linear Predictive Coding (LPC): LPC is one of the potent speech analysis techniques and an assistive method for encoding quality speech at a low bit rate. The primary idea behind LPC is that the particularized speech sample at the prevailing time can be estimated as a linear combination of past speech samples.

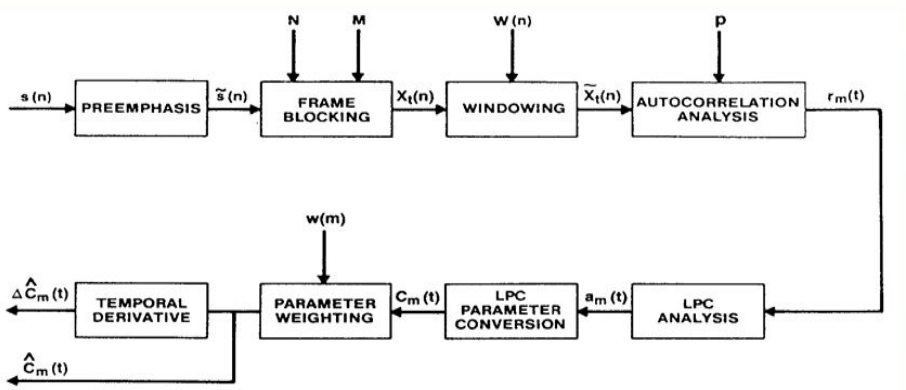


Figure 4: LPC Processor for Speech recognition.

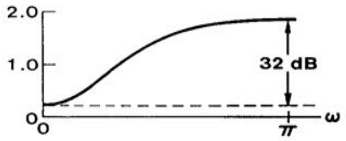


Figure 5: Magnitude Spectrum of LPC, a = 0.95

Figure 4 and 5 shows the block diagram and spectrum plot of LPC. It uses a conventional source filter model, in which the glottal, vocal tract and lip radiation transfer functions are integrated into one all-pole filter that simulates acoustics of the voice vocal tract. The main principle which LPC works on is to minimize the sum of the squared differences between the original speech sample and the approximated speech signal over a known finite duration. This will give the unique combination set of predicted coefficients, which are estimated for every 20ms long frames. The predicted coefficients are given by ak, gain is by G and hence the transfer function of the time varying digital filter is given by

$$H(z) = G/(1-\Sigma kz-k) \quad \text{.....(1)}$$

Where k=1 to p, (10 for LPC algorithm and 18 for improved algorithm). Further Levinsion-Durbin recursion will be used to compute the required parameters for the auto-

correlation method. These cepstral coefficients are LPC's, and the types of such LPC's are Voice-excitation LPC. Residual excitation LPC, Pitch excitation LPC, Multiple excitation LPC, Regular pulse excited LPC and Coded excited LPC [13].

3.1.2 Perceptual Linear Predictive coefficients (PLP's): The main goal of the PLP models are to describe the Psychophysics of people hearing more accurately in the feature extraction process. It is similar to the above-mentioned LPC analysis too and is based on the short-term spectrum of speech. The PLP parameters take advantages of Bark-spaced filter-bank of 18 filters for covering the range of the frequency. Especially the PLP coefficients are evaluated as follows:

- The discrete time-domain input signal is the subject to N-Point DFT.
- The critical band power spectrum is evaluated through DCT of the power spectrum with the piecewise estimation of the critical-band curve, where bank warped frequency procured through the Hertz to Bark conversion.
- Equal loudness pre-emphasis is applied on the down-sampled.
- Intensity-loudness compression will be performed.
- To obtain the equivalent autocorrelation function, the obtained result will be inversed by DFT.
- Finally, the autoregressive coefficients will be converted and modelled to obtain the PLP coefficients [13].

3.1.3 Mel Frequency Cepstral Coefficients (MFCC): The usage of MFCC will be considered as one of the standard tactics for feature extraction owing the reason that it places the frequency bands logarithmically so that it determines the human speech system more intimately than any other methods. Since it has less complex to implement for feature extraction algorithm, only 13 – 16 coefficients of MFCC corresponding to the frequency spectrum of speeches are extracted from the voice or speech sample.

Figure 6 shows the block diagram of MFCC, foster will be the explanations step by step.

1. Pre-emphasis: The speech signal s(n) is an input signal which fed to the high-pass filter: $s_2(n) = s(n) – a*s(n-1)$ where $s_2(n)$ is the output signal and the value of a is usually between 0.9 and 1.0. The z-transform of the filter is,

$$H(z)= 1-a*z^{-1} \quad \text{.....(2)}$$

This step is to analyse the high frequencies. And most of the motivations for the pre-emphasis filter is achieved using mean normalization and the signal plots before and after pre-emphasis are given below (Figure 7).[14]

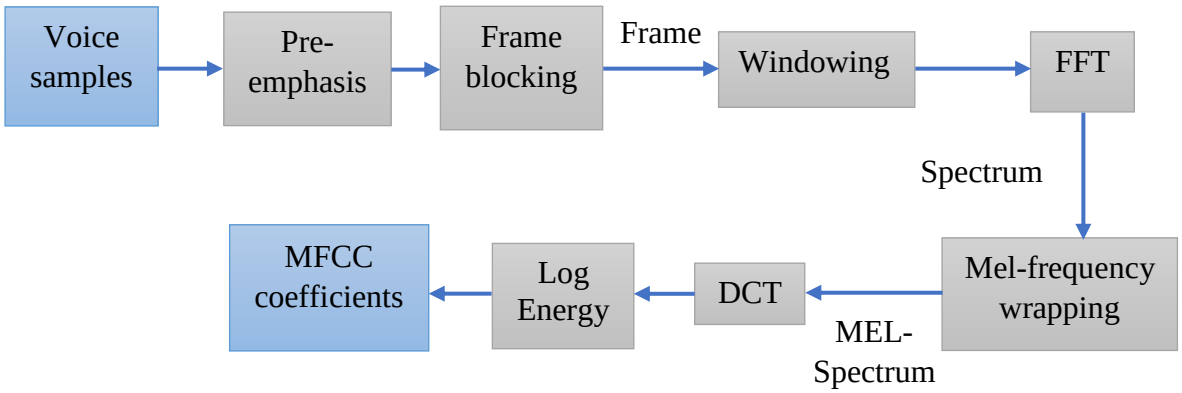


Figure 4: MFCC Block Diagram

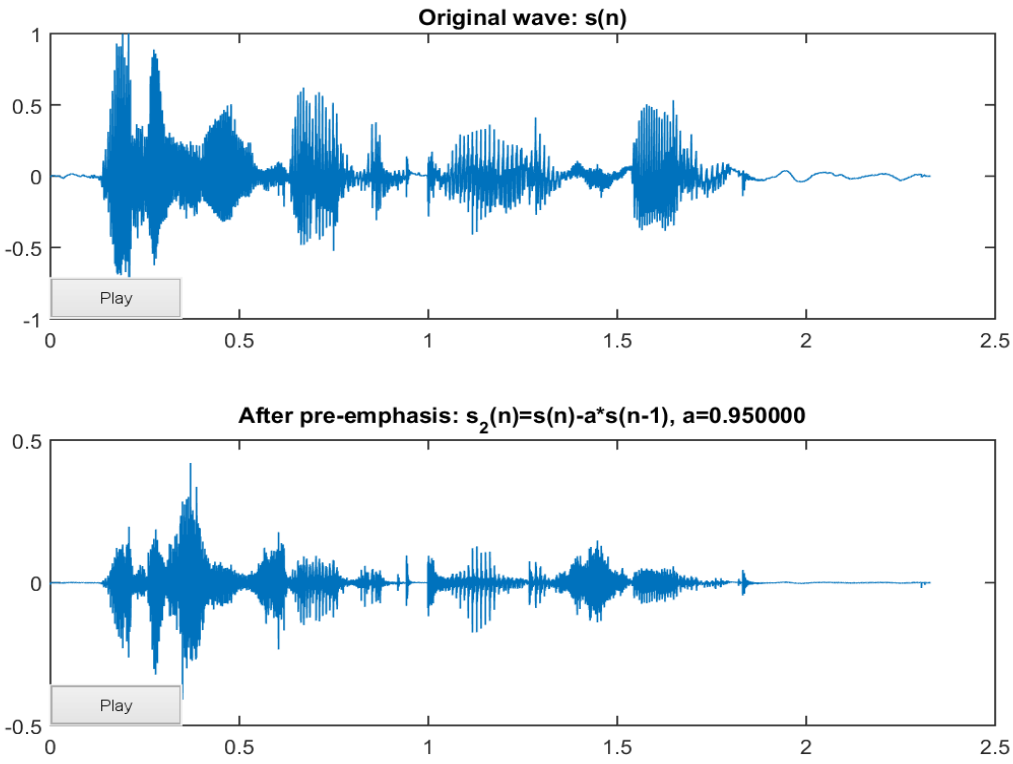


Figure 5: Pre-emphasis Plots

2. Frame Blocking: After pre-emphasis, the complete high frequency signal must be divided into frames and this process is done by framing. The signal is split into short-time frames. The continuous speech signal will be blocked into frames of N samples with outlying frames being separated by M(M<N). The first frame will be consisting of the first N

samples. The second frame being M samples which overlaps the 1st frame by N-M samples. Likely the third frame will be 2M which will be overlapped to 2nd by N-2M samples. This framing process will be continued until all the speech is fitted into within one or more frame blockings.

If the value of N=256, which is equivalent to ~30m sec and M=100, is considered as the compromise between the frequency resolution and time resolution. For N=128, the high-resolution frame will be obtained. Subsequently each frame will have very short period of time and in turn only 65 distinct frequencies samples obtained that means a very poor frequency resolution. If N=512, an excellent frequency resolution will be obtained but results in lesser frames giving the resolution in time is widely reduced. Thus, to conclude N=256, a compromising frame block is best for the fit.

3. Hamming Windowing: The succeeding processing step is windowing. As from the survey information, it is clear that Hamming window is best to use with the process. So hamming window which is also called as raised cosine window is used for the study. In signal processing, window is usually use for a signal which has limited length. That means a window is used when a signal/frame which is under processing has limited length. Indeed, the processing signal has to be finite in time, and a process calculation is possible only when it has finite number of points. Likewise, to minimize the discontinuities of the signal frames at the beginning or at the end of each frame, this process is important. Thus, to maintain a signal in finite time hamming windowing is applied. The spectral distortion by using window to taper the signal to zero when it is necessary. If the window is defined as $w(n)$, $0 \leq n \leq N-1$, where N is the number of samples in each frame, then the result of windowing is the signal.

$$Y_1(n) = x_1(n)\, w(n),\ 0 \leq n \leq N-1 \quad \textbf{.....(3)}$$

Hamming window is represented by the equation,

$$w(n-\alpha) = (1-\alpha) - \alpha \cos(2\pi n/N-1),\ 0 \leq n \leq N-1 \quad \textbf{.....(4)}$$

and for the different values of α gives the different curves for the hamming window which is showed in the figure 8.

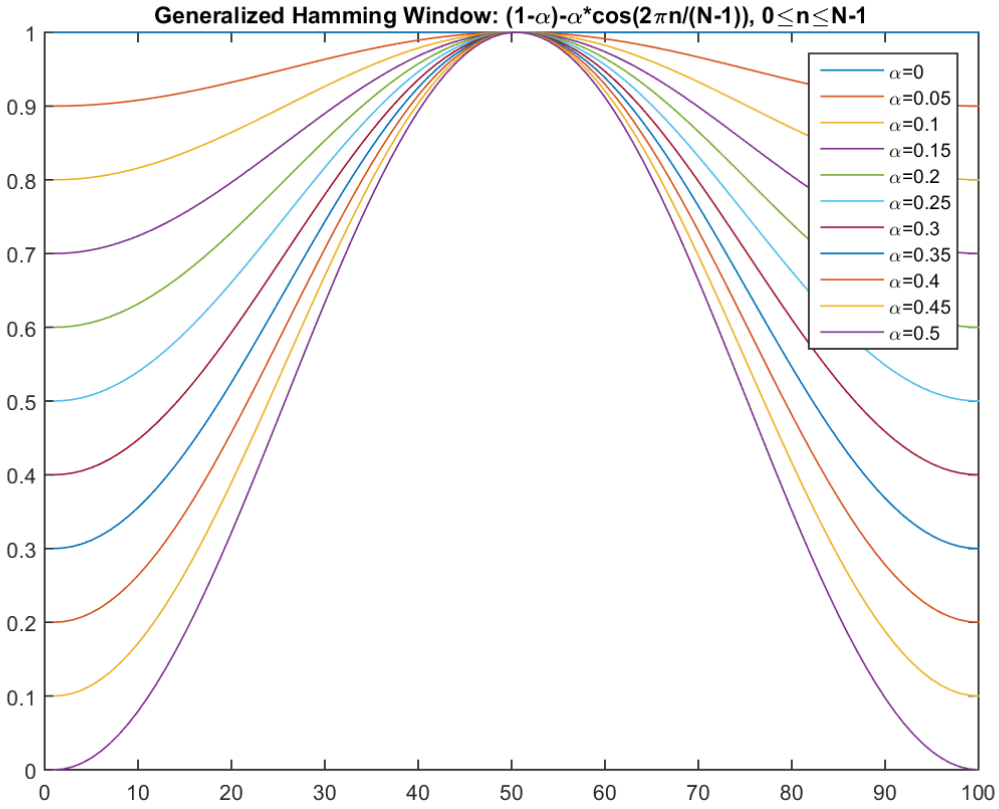


Figure 6: Hamming window

4. Fast-Fourier Transform- FFT: The present signal/sample is in time domain after all the previous processes. So, in order to convert to frequency domain, every frame of N sample, this process will be carried out. And also, the vocal tract impulse response and the convolution of glottal pulse will be converted to obtain the magnitude response of each frame.

   When FFT is performed on a frame, it is assumed that the signal with in a frame is periodic and continuous wrapping around. If it's not this case, then the FFT is still perform but the in-continuity at the frames beginning and the final points can be acquainted by undesirable effects in the frequency response. To clear this issue, there are two approaches:

   a. Multiply every frame by the previous hamming window to increase its continuity in the beginning and end points.
   b. And to take a frame whose size is equal to variable size, such that it always contains an integer multiple number of the primal periods of the voice sample.

   Implementing strategies like this are not a big problem unless it a voice speech.

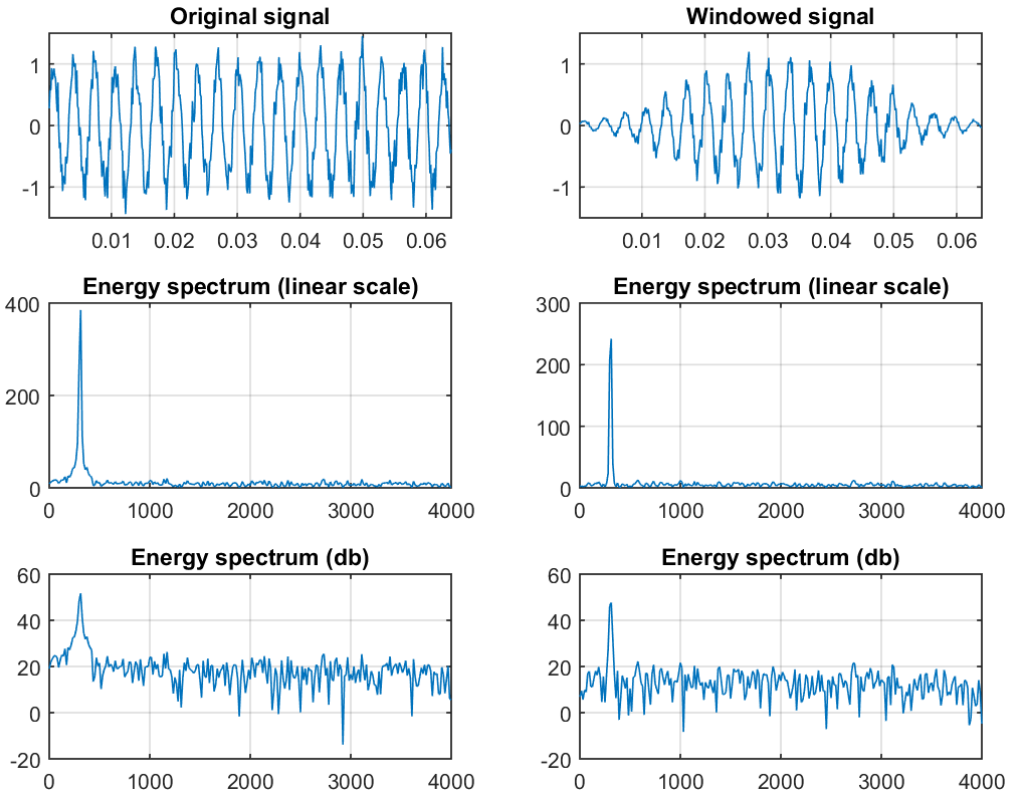


Figure 7: An example of Hamming window (Sinusoidal + noise)

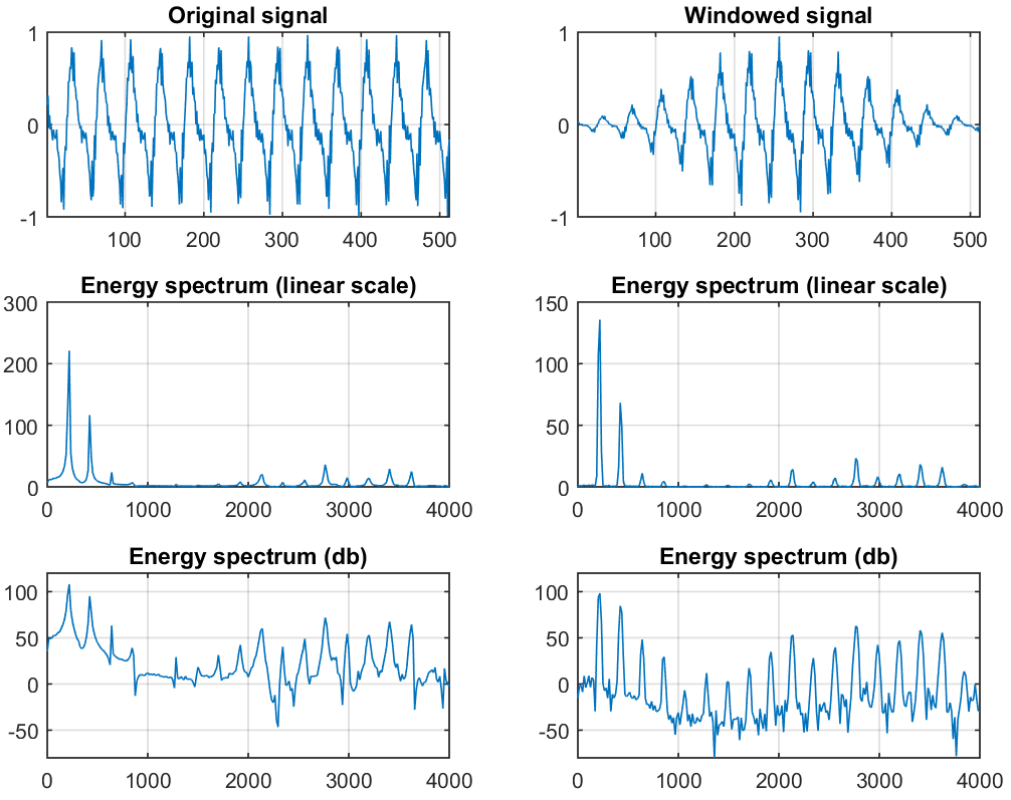


Figure 8: An example of Hamming window (Singing voice's signal)

In the figure 9, the signal sample is a sinusoidal function additionally some noise. As the figures depicts, without the usage of window (Hamming), the discontinuity at the frame's first and last points has made the peak in the frequency response large/wider and little obvious. And in the further figures, with the use of hamming window, the frequency response peaks are sharper and more distinct. To make the differences to depict the acts, the next figure 10 which shows the speech signal for the same excises: And one can observe, with the use of hamming window, the responses are much sharper when compared to original signal since it is a sound clip of a singing voice. In some random cases, if the input frame consists of 3 congruent primal periods, the magnitude response then will be added 2 zeros between each 2 neighbouring points of the frequency response. However, the required information will be inside the frame, thus, to extract the envelop-like-features, the next process triangular bandpass filters will be applied.[14]. In other words, theoretically, FFT which is the fast algorithm to implement DFT which can be defined and symbolizes as on the set of N samples is as follows:

$$X_n = \sum_{k=0}^{N-1} x_k \, e^{-\frac{2\pi jkn}{N}}, \quad n = 0,1,2 \dots \dots, N-1 \quad \text{.....(5)}$$

Where j is an imaginary unit.

The result of this process is also called as or often represented as Spectrum or Periodogram.

5. Triangular Bandpass filters / Mel frequency wrapping: The magnitude response of the sample will next be multiplied by a set of 20 triangular bandpass filters to get log energy for each of the triangular bandpass filter. The positions of this filters will be equally spaced according to the mel frequency. Since the frequency components does not follow the linear scale, a subjective pitch is measured on a scale called 'mel' scale with each tone of an actual frequency. The mel-frequency scale is a linear frequency which is spaced lower than 1000 Hz and above the logarithmic space of 1000 Hz. Therefor the following formula can be used approximately for the purpose.

$$f_{MEL} = 1125 * log_{10}(1 + {}^{f_{HZ}}/_{700}) \quad \text{.....(6)}$$

The number of mel spectrum coefficients K is normally chosen as 20. Figure 11 shows the typical filter bank when it is applied to the frequency domain. And the relationships between the mel and the linear frequencies are also showed in figure 12.

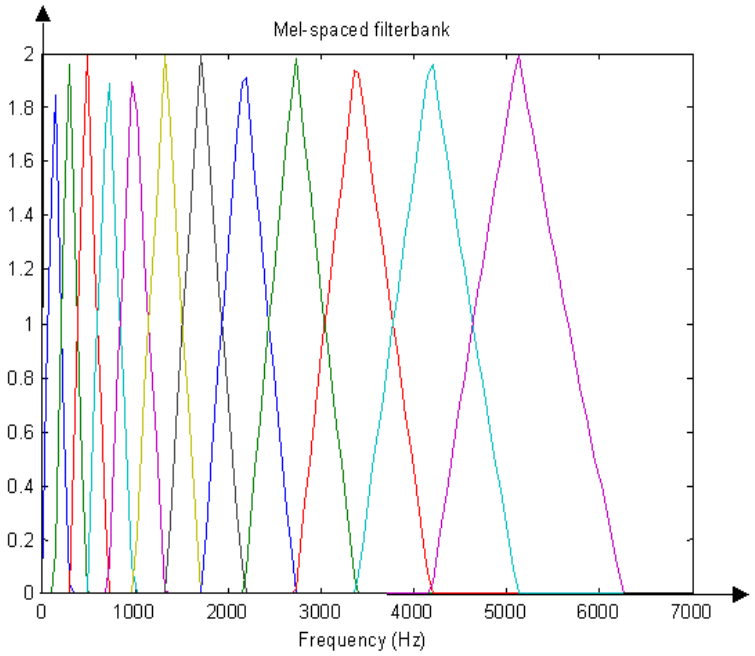


Figure 9: Mel-spaced filter-bank for 20 filters

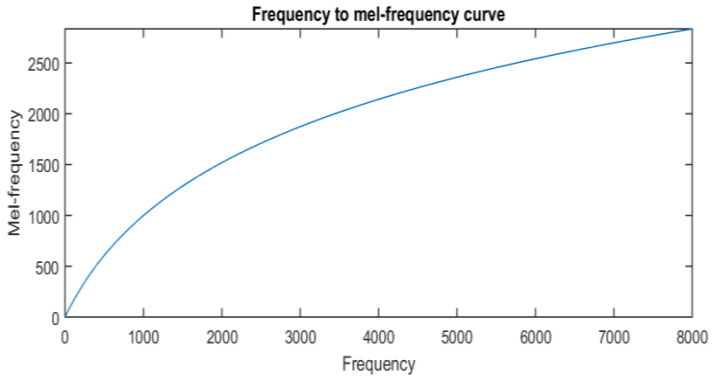


Figure 10: Plot between mel and linear frequencies

6. <u>Discrete cosine transform – DCT</u>: DCT which is called as Cepstrum, is the inverse Fourier transform of the logarithm of the power spectrum of a signal. It is also related to the auto correlation function and separates the glottal frequency from the vocal tract resonances. In this pace, DCT is applied on the 20log energy Ek which is obtained from the previous process, i.e., triangular bandpass filters and will have L mel-scale cepstral coefficients, and N number of triangular bandpass filters. The equation of this function is given by,

$$C_m = \sum_{k=1}^{N} Cos\left[m * (k - 0.5) * \frac{\pi}{N}\right] * E_k \quad , \quad m = 1,2,3, \dots \dots L \quad \textbf{……..…..(7)}$$

Since FFT and DCT transforms are done, the frequency domain to time domain change is called as quefrency domain. The obtained resulted features of this cepstrum are referred as the mel-scale cepstral coefficients or MFCC. For better performance, the log energy will be performed, in next step.

7. Log energy: The energy for the frame or within the frame is also easy to get from the process, as the addition of log energy as the 13th feature to MFCC. If there is a necessary, some other features like step, pitch, zero cross rate and high-order spectrum momentum and so on can also be added here after.[14]

3.2 Feature Classification or Selection: Feature classification is applied on the extracted features from the MFCC in this context. Foremost than the classification, approaches should explain first, so basically approaches for the speaker recognition are

- Acoustic Phonetic approach: This approach is broadly characterized by group of acoustic properties that are evidenced in the speech signal over time. The main step in the acoustic phonetic approach is that a speech spectral analysis together with feature detection, converts the spectral measurements into a set of features. Those features describe the wide acoustic properties of the various phonetic units. And the further steps follow like, segmentation, labelling phase, phonetic labels characterization of the speech and attempts to determine a string from the labels sequences.[15]
- Pattern recognition approach: This approach involves 2 fundamental steps, pattern training and pattern comparison. A speech signals pattern can be in the type of a speech template or a statistical model. Thus; there exists the two main sub approaches of pattern recognition approach. There are template approach and stochastic approach.
  - Template approach: The principal idea for this approach is simple, by creating a collection of prototype speech patterned templates and save them as reference pattern. Then recognition is carried out by matching an unknown speech of the different utterances, like one versus one verification. The important keypoint here is to use dynamic programming to temporarily align patterns to account for differences.
  - Stochastic Approach: This is the very important part of this research work, because main classification parts are explained. Stochastic modelling implies

the use of probabilistic models to grapple with incomplete and uncertain information. As the uncertainty and incompleteness discussed in the above section, after the extraction process, those uncertainties will be dealt here. Considering an example, speaker's variability, contextual effects and homophones words. In these cases, this approach is the suitable solution to the speaker recognition.[15]. The few popular stochastic approached models are Hidden Markov modelling, Dynamic time warping, Vector quantization, Identity vector, Support vector machines, Joint factor analysis, Gaussian mixture modelling, Maximum A posteriori adaptation etc.,

3.2.1 **Vector Quantization:** One of the most convenient unsupervised learning method for speakers features classification is Vector Quantization. After the speech feature extraction, the extracted feature space is partitioned into a set of mutually privileged convex regions. Results in having the lesser number of features compared to the original feature space. Thus, it is a process of mapping vectors from a big vector space to a finite small number of regions in the space[16], This every region is called as cluster and this cluster is represented by a centre is referred as centroid/codeword. Furthermore, the collection of this all codewords is termed as codebook, which is the output of the VQ i.e., Vector codebook.

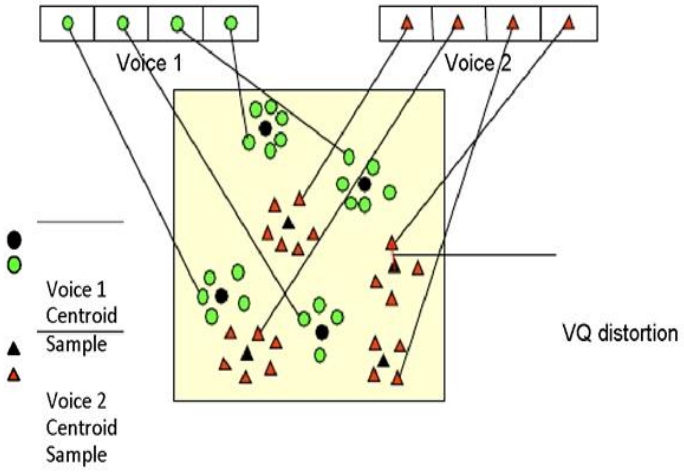


Figure 11: Conceptual diagram illustrating Vector Quantization Codebook formation

The above figure 13 depicts the representation of vector codebook formation in the recognition process when there is only two speakers and two dimensions of the acoustic space. Circles refers to the speaker 1 and the triangle refers to speaker 2. In the training phase, a speaker specific codebook will be generated for every known speaker. The black circles and triangles in the figure exemplify the resulted centroids which are formed after quantization.

The distance from one of the vector and the nearest centroid is addressed as VQ-distortion. The recognition part of the process is not applied here because vector quantization is used only for classification and total VQ distortion is not applicable. But further to explain the algorithm, the flow continues.

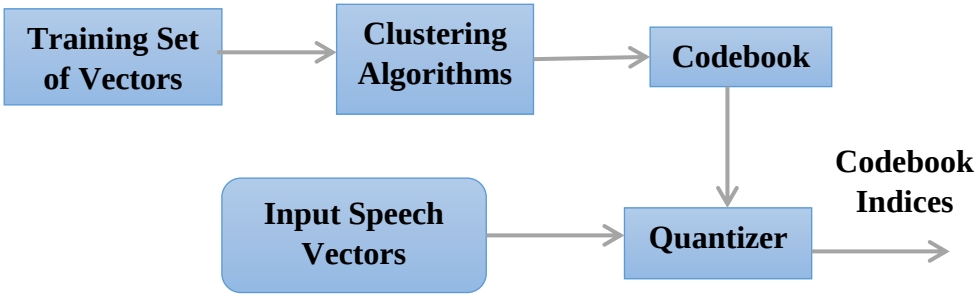


Figure 12: Block Diagram of the basic VQ training and classification structure

Clustering of the training vectors: After the enrolment briefing, the acoustic vectors extracted from the input speech from each speaker on the training vectors. Then codebook is created for each speaker using those training vectors. Linde, Buzo and Gray, LBG algorithm for this clustering a set of L training vectors into M codebook vectors will be used. This algorithm implements an M-vector codebook in number of stages. The initial step is by designing a 1-vector codebook, then by the splitting technique from the centroids to begin the search for a 2-vector codebook and proceeds the splitting process until to obtain the desired M-vector codebook.

LBG algorithm (Figure 15) is implemented by the preceding recursive procedure.

i. Using the centroid of the entire training set, design a 1-Vector codebook.
ii. By splitting every presented codebook y_n, double the size of the codebook, according to the rule:

$$y_n^+ = y_n(1+\epsilon) \quad \text{.....(8)}$$

$$y_n^- = y_n(1-\epsilon) \quad \text{.....(9)}$$

Where n varies from 1 to the current size of the codebook, and ϵ is a splitting parameter.

iii. Nearest-Neighbour search: find the centroid which is closest from each of the training vector and designate that vector to the corresponding cell.
iv. Centroid Update: For every cell, update the centroid using the training vectors centroid which is assigned to the cell.

v. Iterations 1 & 2 : until the average distance falls below a pre-set threshold repeat the steps 3 and 4 and for the codebook size of M , repeat the steps 2, 3 and 4 [16].

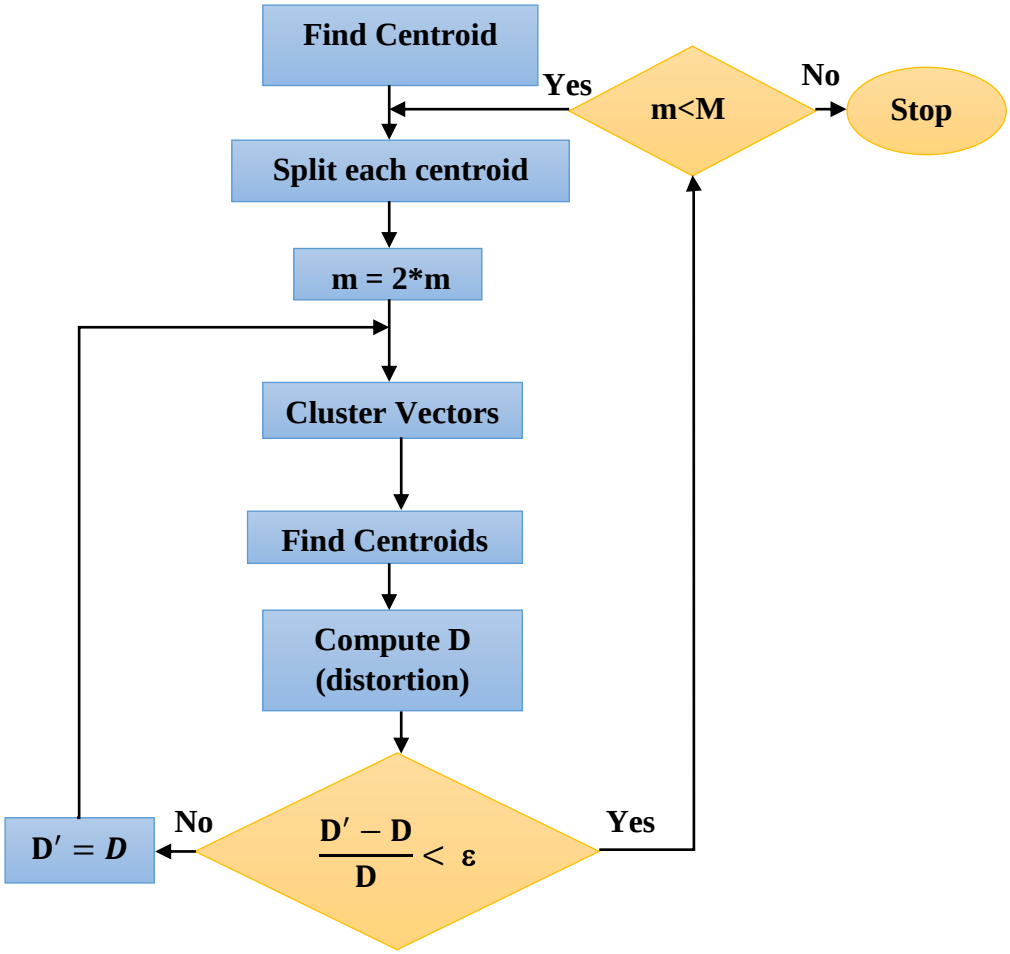


Figure 13: LBG Algorithm Flowchart

3.2.2 **Identity Vector:** The most popular approaches which are based on GMM-UBM are JFA and Identity Vectors. In JFA, its modelling is defined by two distinct spaces: the speaker's calculated space is defined by the eigen-voice matrix and the channel space is portrayed by eigen-channel matrix. But unless like this method, instead of declaring two separate spaces, this new method defines only a single space. In this new space, a new vector represents the given speech recording called total factors and this framework is i-vector and it is in state-of-the-art in the field.

For a given utterance, the channel and speaker dependent GMM supervector is defined as follows:

$$M = m + Tw \qquad \text{.....(10)}$$

Where m is an independent of speaker and channel supervector, T is a low rank rectangular matrix and w is a standard normal distribution random vector N (0,1) and the components

of the vector w are the total factors. These new vectors are called i-vectors. M is to be normally distributed with mean vector and covariance matrix TT^t. The total factor is a hidden variable, which can be defined by its posterior distribution conditioned to the Baum-Welch statistics for a given utterance. This posterior distribution is a gaussian models and the mean of this corresponds to the i-vector. Thus, UBM are extracted using Baum-Welch statistics. The given sequence of the L frames $\{y_1, y_2, \ldots\ldots y_n\}$ and the UBM Ω which is composed of C matrix components defined in the feature space. Hence, the Baum-Welch statistics have to estimate the i-vector for a given speech utterance u is given by:

$$N_c = \sum_{t=1}^{L} P(c|y_t, \Omega) \quad \text{.....(11)}$$

$$F_c = \sum_{t=1}^{L} P(c|y_t, \Omega) y_t \quad \text{.....(12)}$$

Where c = 1, 2 ,……C is the Gaussian index and $P(c|y_t, \Omega)$ corresponds to the posterior probability of mixture component c generating the vector y_t. Then it followed to the centralized first-order Baum-Welch statistics has also to be computed for the extraction of i-vectors as follows:

$$\hat{F}_c = \sum_{t=1}^{L} P(c|y_t, \Omega)(y_t - m_c) \quad \text{.....(13)}$$

Where m_c is the mean of UBM mixture component c. then the equation of the i-vector for a given utterance can be obtained by:

$$w = (I + T^t \sum^{-1} N(u)T)^{-1} . T^t \sum^{-1} \hat{F}(u) \quad \text{.....(14)}$$

Where N(u) is a diagonal matrix. The supervector which is obtained by concatenating all first-order Baum-Welch statistics Fc for a given utterance u can be represent as $\hat{F}(u)$ which has CFx1 dimension. After extraction of raw i-vectors to remove the useless information normalization should be carried out.

- Artificial intelligence approach: This approach is a hybrid combination of acoustic phonetic and pattern recognition. Knowledge based approach uses the information regarding phonetic, linguistic and spectrogram. This knowledge is derived from the spectrograms and study of spectrograms and is included with the procedures. However, this approach is with limited success because of its quantifying expert knowledge but often used only to work on algorithms to make them better. [15]

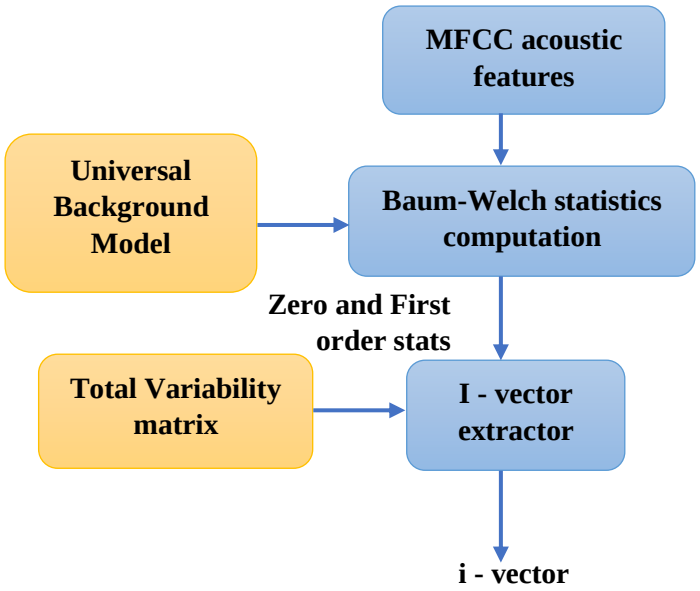


Figure 14: Identity vector extraction process diagram

3.3 Feature Matching: This section is the last processing step which returns the result of the whole process. Generally, in speaker identification, there will be only two major modules, feature extraction and matching, but in this context feature extraction, feature classification/normalization and feature scoring/ matching is conducted to give more precise results with new methodologies, and also, it is referred for the speaker verification. Thus, the following methods will be undertaken for this verification:

3.3.1 **Probabilistic Linear Discriminant Analysis (PLDA):** PLDA is unlike the i-vectors, state-of-the-art in speaker verification systems and the extended part of the i-vector in the form to split the total data variability into within-individual and between individual variabilities. Here in this context, instead of original PLDA formulation, the two alternative variants which are equal to full covariance are considered. Simplified PLDA and two-covariance model but considering one more PLDA model i.e., Standard PLDA is also explained. [17]

The three types of PLDA model's unified formulation and its variants:

i. Standard PLDA: It is defined as

$$\Phi_{ij} = \mu + Vy_i + UX_{ij} + \varepsilon_{ij}, \quad \text{.....(15)}$$

$$y_i \sim N(0, I), \quad \text{.....(16)}$$

$$x_{ij} \sim N(0, I), \quad \text{.....(17)}$$

$$\varepsilon_{ij} \sim N(0, \Lambda^{-1}), \quad \text{.....(18)}$$

Where $\Phi \in R^{Dx1}$,$\wedge$ is a diagonal precision matrix, μ is a global mean.

ii. Simplified PLDA: It is defined as,

$$\Phi_{ij} = \mu + Sy_i + \varepsilon_{ij}\,, \quad \text{.....(19)}$$

$$y_i \sim N(0, I), \quad \text{.....(20)}$$

$$\varepsilon_{ij} \sim N(0,\ \wedge_f^{-1}), \quad \text{.....(21)}$$

Where $\wedge^f$ is a full precision matrix instead of the diagonal matrix.

iii. Two-covariance model: This is defined as,

$$y_i \sim N(y_i\,|\mu, B^{-1}), \quad \text{.....(21)}$$

$$\Phi_{ij}|\ y_i \sim N(\Phi_{ij}|y_i\,, W^{-1}), \quad \text{.....(22)}$$

Where both B and W are full precision matrices.

Exploring the structure of the models: Since the latent variables have a Gaussian distribution, its observed variables also is a gaussian:

$$\Phi_{ij}\big|\ y_i\,, x_{ij} \sim N\big(\Phi_{ij}\big|\mu + Vy_i + UX_{ij}, \wedge^{-1}\big), \quad \text{.....(23)}$$

And also, an integration of the channel latent variable leads to a result in the closed form as:

$$\Phi_{ij}|\ y_i \sim N(\Phi_{ij}|\mu +\ Vy_i + UU^T\ + \wedge^{-1}, \quad \text{.....(24)}$$

Now, formulate the above equation in a similar way to two-covariance model:

$$\widehat{y_i} \sim N(\widehat{y_i}\,|\mu, VV^T), \quad \text{.....(25)}$$

$$\Phi_{ij}|\ \widehat{y_i} \sim N(\Phi_{ij}|\widehat{y_i}\,, UU^T\ + \wedge^{-1}\,, \quad \text{.....(26)}$$

Comparing the equations 25 to 21 and 26 to 22, says that the structure of a standard PLDA and a two-covariance model is identical and their only difference is the covariance matrices. If to consider the individual covariance matrices,

$$W_3^{-1} = W^{-1}, \quad \text{.....(27)}$$

$$B_3^{-1} =\ B^{-1}, \quad \text{.....(28)}$$

Thus, the resultant will be from 25 and 26 that,

$$W_1^{-1} = UU^T + \wedge^{-1}, \quad \text{.....(29)}$$

$$B_1^{-1} = VV^T, \quad \text{.....(30)}$$

Hence finally, applying the same analysis to the simplified PLDA leads to the next equations.

$$W_2^{-1} = \Lambda_f^{-1}, \quad \text{.....(31)}$$

$$B_2^{-1} = SS^T, \quad \text{.....(32)}$$

$$B_1^{-1} = VV^T = V(RR^T)V^T = (VR)(VR)^T, \quad \text{.....(33)}$$

$$VV^T = LL^T \quad \text{.....(34)}$$

Scoring: At verification step, a pair of individual models are taken, one formed from the enrolled features and the other from the test features of the claimed person, thereby decision should take whether these models belongs to same person or not. To do so, log-likelihood ratio also can be calculated.[17]

EM-Algorithms: The main drawback of this algorithm is that at the E-step, matrix needs to be inverted whose size grows linearly with the number of samples per person. And hence, this implementation became highly impractical for large datasets. But from other researchers, a special matrix structure PLDA model is derived and is presented in the below figure 17 (Algorithm 1) and the incomplete algorithm (only E-step) for the two-covariance model is presented in figure 18(Algorithm 2). [17]

Algorithm 1: Scalable PLDA learning algorithm

Input: $\mathbf{\Phi} = \{\phi_{ij}\}_{i=1,j=1}^{K,n_i}$, where K is a total number of persons, and n_i is the number of samples for i-th person.
Output: Estimated matrices $\mathbf{V}$, $\mathbf{U}$ and $\mathbf{\Lambda}$.
Sort persons according to the number of samples $\{n_i\}$;
Find total number of samples N and center the data (eq. A.1 and A.2) ;
Compute data statistics $\{\mathbf{f}_i\}$ and $\mathbf{S}$ (eq. A.3 and A.4) ;
Initialize $\mathbf{V}$ and $\mathbf{U}$ with small random values, $\mathbf{\Lambda} \leftarrow N\mathbf{S}^{-1}$;
repeat
  E-step:
    Set $\mathbf{R} \leftarrow 0$;
    Compute auxiliary matrices $\mathbf{Q}$, $\mathbf{J}$ (eq. A.5 and A.6) ;
    for $i = 1$ *to* K **do**
      if $n_i \neq n_{i-1}$ **then** compute $\mathbf{M}_i$ (eq. A.7);
      else $\mathbf{M}_i \leftarrow \mathbf{M}_{i-1}$;
      Find $\mathbb{E}[\mathbf{y}_i]$ (eq. A.8) ;
      Update $\mathbf{R}_{\mathbf{yy}}$ (eq. A.13);
    Calculate $\mathbf{T}$, $\mathbf{R}_{\mathbf{yx}}$ and $\mathbf{R}_{\mathbf{xx}}$ (eq. A.12, A.14 and A.15) ;
  M-step:
    Find $\mathbf{V}$, $\mathbf{U}$, $\mathbf{\Lambda}$ (eq. A.16 and A.17) ;
  MD-step:
    Compute auxiliary matrices $\boldsymbol{\mathcal{Y}}$, $\mathbf{G}$, $\boldsymbol{\mathcal{X}}$ (eq. A.18, A.19 and A.20) ;
    Update $\mathbf{U}$, $\mathbf{V}$ (eq. A.21 and A.22) ;
until Convergence ;

Figure 15:Algorithm 1 for Inverse matrices

Algorithm 2: Two-covariance model learning algorithm

Input: $\mathbf{\Phi} = \{\phi_{ij}\}_{i=1,j=1}^{K,n_i}$, where K is a total number of persons, and n_i is a number of samples for i-th person.

Output: Estimated matrices $\boldsymbol{\mu}$, $\mathbf{B}$ and $\mathbf{W}$.

Sort persons according to the number of samples $\{n_i\}$;

Compute data statistics N, $\{\mathbf{f}_i\}$ and $\mathbf{S}$ (eq. B.1, B.2 and B.3) ;

Initialize $\boldsymbol{\mu}$, $\mathbf{B}$, $\mathbf{W}$;

repeat

  E-step:

    Set $\mathbf{T} \leftarrow 0$, $\mathbf{R} \leftarrow 0$, $\boldsymbol{\mathcal{Y}} \leftarrow 0$;

    for $i = 1$ *to* K **do**

      if $n_i \neq n_{i-1}$ **then** compute $\mathbf{L}_i$ (eq. B.4);

      else $\mathbf{L}_i \leftarrow \mathbf{L}_{i-1}$;

      Find $\mathbb{E}[\mathbf{y}_i]$ and $\mathbb{E}[\mathbf{y}_i\mathbf{y}_i^{\mathsf{T}}]$ (eq. B.5, B.6) ;

      Update $\mathbf{T}$, $\mathbf{R}$ and $\boldsymbol{\mathcal{Y}}$ (eq. B.8, B.9 and B.10);

  M-step:

    Find $\boldsymbol{\mu}$, $\mathbf{B}$ and $\mathbf{W}$ (eq. B.11, B.12 and B.13) ;

until Convergence ;

Figure 16: Algorithm 2 for two-covariance matrices

3.3.2 **<u>Euclidean Distance:</u>** As the Euclidean Distance associated with the vector quantization function, depending upon the size of the VQ codebook and the training patters are taken from the code vectors that makeup a codebook. Thus, Euclidean Distance is evaluated between centroids and cepstrum. d(x,p) is the minimum of Euclidean Distance where it gives the pairwise Euclidean Distance between columns of two matrixes. The result will be recorded for the future comparisons. The measured Euclidean Distance is the standard distance measure between two vectors. Theoretically, to measure the Euclidean Distance, computation must be done as the sum of the squares of the differences between the individual components of x and p and is represented as the second equation.

$$d(x,p) = \sqrt{\sum_{i=1} (xi - pi)^2} \quad , \qquad \text{.....(35)}$$

$$d(x,p) = (xi - pi).(xi - pi)' \qquad \text{.....(36)}$$

3.3.3 **<u>Support Vector Machine (SVM):</u>** A sparse kernel decision machine classifier which runs for classification, is the best to choose SVM. This approach to the

principled problems is good because of its mathematical foundation in statistical learning theory and in terms of a subset of the training input, it creates its solution. This algorithm has been considerably used for regression, classification, novelty detection tasks and feature reductions. But in this context, it is used for supervised Classification tasks.

From a geometric perspective, learning a classifier is corresponds to generate the equation for a multidimensional surface that best separates the different classes in the feature space. This is a discriminant technique and genetic algorithms and perceptron's are widely used for classification. The goal of genetic algorithm and perceptron is to minimize error during training. This in turn translate into many hyperplane's which are meeting its requirement. Figure 19 depicts the multiple hyperplanes obtained by SVM, perceptron and GA classifiers on two-dimensional, two class data. The points which are surrounded by circles illustrates the support vector.

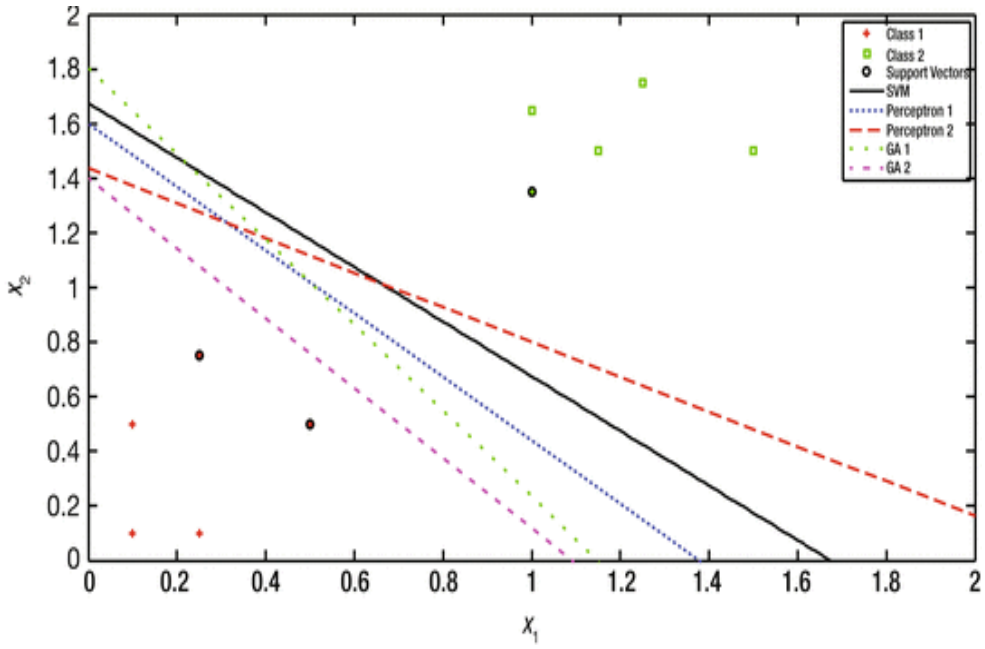


Figure 17:: Two-dimensional, two-class plot for SVM, Perceptron and GA hyperplanes

Few highlighted properties of machine learning SVM are as follows:

1. SVM is a sparse technique: when the parameters of the model are learned it stores all the training data in the memory. However, if once model parameters are identified, then SVM counts on only on a subset of those trained vectors called support vectors for future prediction. Using the Lagrangian relaxation, it creates a hyperplane in such a way that an optimization step should involves an objective

function regularized by an error term and a constraint. The original dataset obtained is data dependent and varies based on the data complexity, which is snatched by the data dimensionality and class separability.

2. SVM is a kernel technique: to map the data into a higher dimensional space, SVM use kernel trick. This is before solving the machine learning task as a convex optimization problem. To learn a nonlinear separating boundary increases the computational requirements during the optimization phase. Instead using the predefined kernel function SVM maps the data into a new and higher dimensional space.
3. SVM is a Maximum Margin Separator: The hyperplane has to be situated in such a way that it is at a maximum distance from the different classes. And also, it minimizes the error and cost function. This is s necessity because training is done on a sample of the population, whereas prediction has to calculated on approximation instances that might have distribution that is little different from that of the subset which is trained.
4. SVM uses structural risk minimization and satisfies the duality and convexity requirements. Figure 20 illustrates the overall model error varies with the complexity index of a machine learning model. From the plot figure it is clearly explainable that as the complexity index increases, the minimum will be the error for the optimal model indexed.[18]

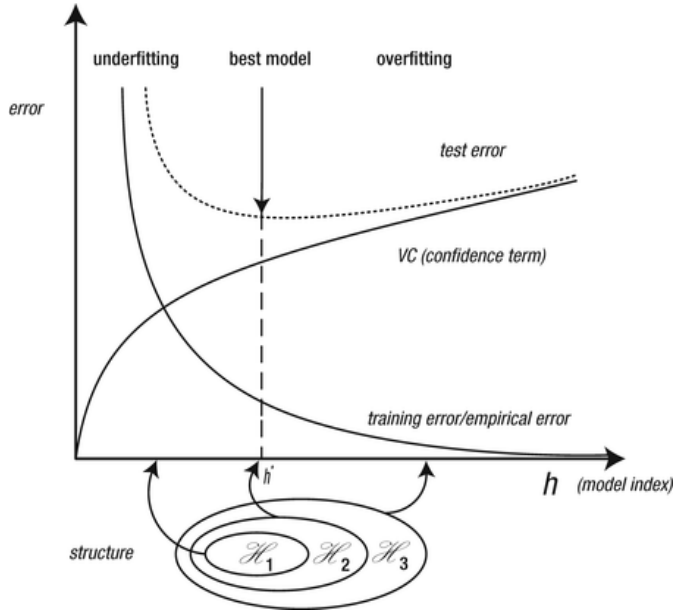


Figure 18: Relationships between error trends and model index

❖ **Hard-Margin SVM:** As it is mentioned, SVM is a technique which classifies based on the hyperplane and a function $g(x) = W^T x + b$ that exactly separates two classes with a maximum margin. Figure 21 shows a separating hyperplane corresponding to a hard-margin SVM.

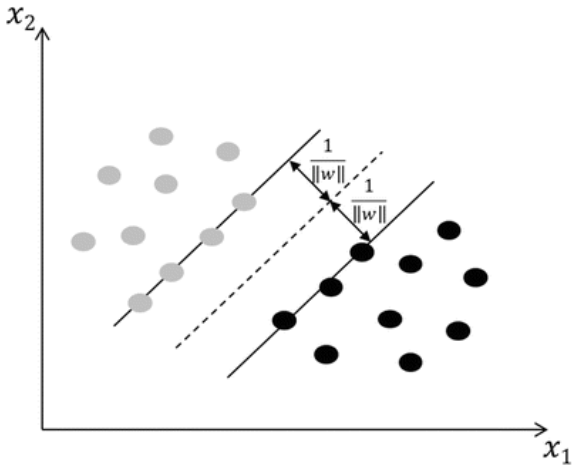


Figure 19: Hyperplane with Maximum margin (SVM)

❖ **Soft-Margin SVM:** When the data are not completely separable as shown in the figure 22, marked with the points X is a slack variable which are adopted to the SVM objective function. In this case, SVM does not search for hard margin instead it will classify smoothly. Now a soft classifier is classifying the data correctly.

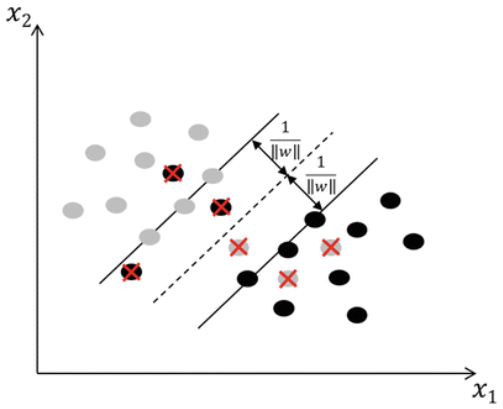


Figure 20: A depict of Soft margin Classifier

❖ **Kernel SVM:** When the data on the plane is non-linear, kernel based SVM is the best to use. In this Kernel transform the data to a higher-dimensional space which is termed as kernel space, in which data is separable linearly. Some popular kernel functions include linear kernel, Polynomial function, Hyperbolic tangent, Gaussian radial basis function, Laplacian radial basis function, Randomized blocks analysis of variance kernel and Linear spline kernel in 1D. For example, figure 23 displays the two

dimensional exclusive OR data, a linearly non-separable feature space. In the latter, when kernel is applied 16 points are created from the 4 inputs. [18]

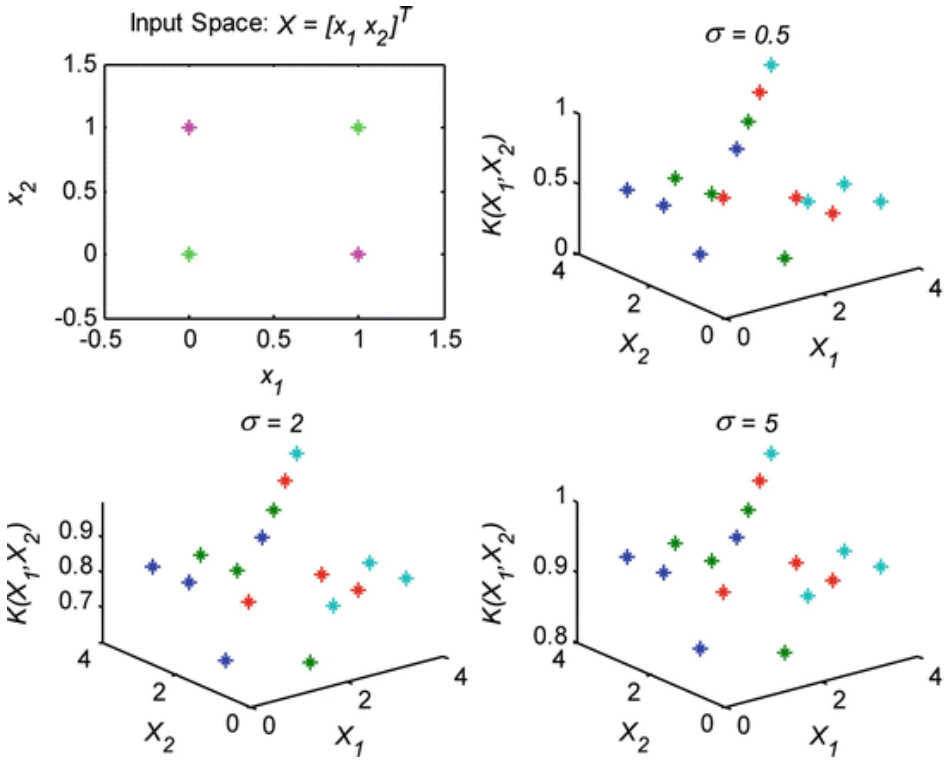


Figure 21:An example of Kernel SVM

- **Multiclass SVM:** The expansion of the binary classification to the multiclass case includes one-versus-rest/ one-against-all (OAA) and the pair-wise classification/ one-against-one (OAO). OAA is the most used multiclass SVM's. It constructs binary classifiers and each classifier differentiates one class from all the rest in-turn induce the two-class problem. Whereas OAO builds binary classifier in such a way that it should distinguish two of the classes only and should evaluate the classifier. A graphical representation of a single SVM and a MSVM is illustrated in figure 24.

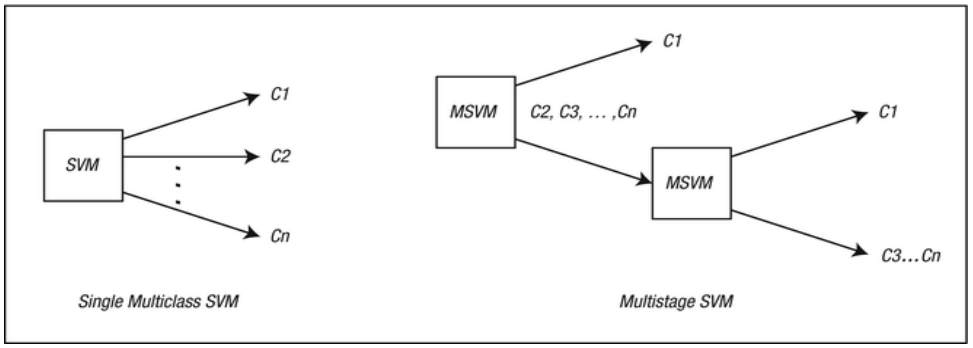


Figure 22: Graphical representation of single SVM and multi SVM's

The multiclassification objective is also mostly works in the same way as OAA technique and it is the most compact form of optimization. The multiclassification objection function constructs two class rules and decision function solve the constraints. When the SVM parametric model allows for adjustments, parameters do not fit the entire dataset. Thus, to solve this problem, partitioning of data into subgroups with similar features is the best solution and to derive the classifier parameters separately. And this overall process results in a multistage SVM or a hierarchical SVM which gives the best accuracy and also induce the likelihood ratio.[18]

4. PERFORMANCE MEASURES

This section explains the performance measures which results from all the mentioned tasks. These measures are important to note because to evaluate the system performance, to know the error and to correct it these are the main key points. In such cases for this context, for Speaker Identification, EER, Thresholding (FAR and FRR), DCF, DET and ROC and for Voice impairments detection Accuracy, Specificity, Sensitivity, Precision, Recall, F-measure, True positive, True negative, False positive and False negative are considered. Further will be the brief explanation.

4.1 Thresholding (FAR and FRR): False acceptance and False rejection rate: The scores or the weights to extract the resemblance between a pattern or the final templates or vectors. The higher the score is, the greater the similarity within them. Theoretically, client score should always be higher than the scores of impostors. Due to few reasons, this can't be true in real and in some cases impostors scores will be higher than the client score. For these reasons, however the classification threshold can be considered, some error will occur.

Depending on the preferred classification threshold, the threshold depending fraction of the falsely accepted patterns divided by the number of all imposter patterns in referred as False Acceptance Rate. Its score is one if all imposter patterns are falsely accepted else zero for the true rejections (Figure 25a). Similar to the imposter patterns, the clients scores also will vary around a certain mean value. If a high classification threshold is applied, some client's patters may falsely get rejected. Depending on the value of the threshold, the fraction of the number of falsely rejected client patterns divided by the total number of client patterns is symbolizes as False Rejection Rate. According to FAR, its value also lies in between zero and one (Figure 25b).[19] These two errors can be explained in equation as follows, where q and r are the claimed speaker model and the UBM respectively.

$$\Lambda(p;q,r) = \int_x p(x) log \frac{q(x)}{r(x)} dx \quad \text{.....(37)}$$

$$\Lambda(p;q,r) = \int_x q(x) log \frac{q(x)}{r(x)} dx \quad \text{.....(38)}$$

$$\Lambda(p;q,r) = \int_x r(x) log \frac{q(x)}{r(x)} dx \quad \text{.....(39)}$$

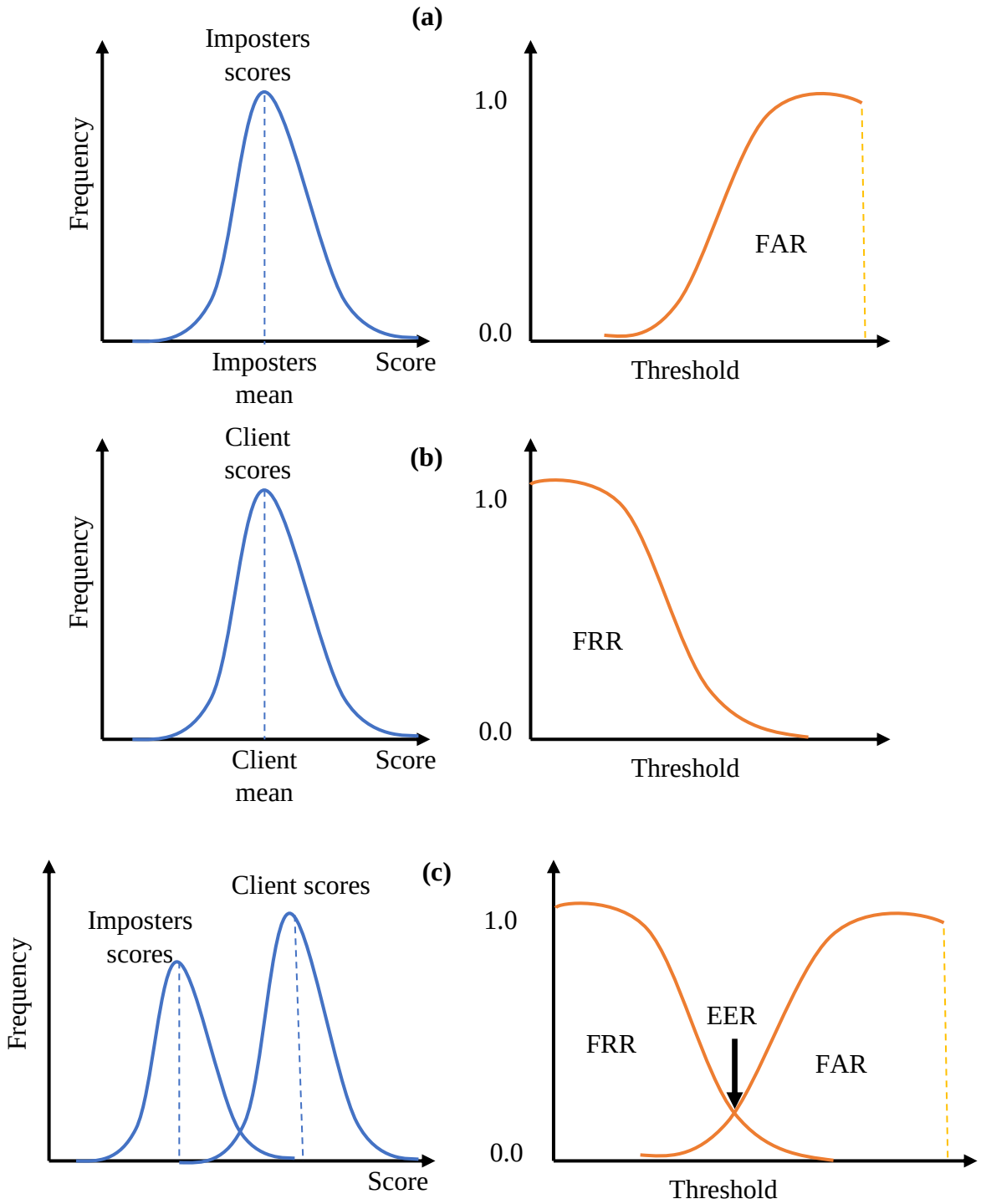


Figure 23: FAR, FRR and EER graphical Plots

4.2 Equal Error Rate: The score distribution overlap at a point where FAR and FRR intersects each other at a certain point. The value of FRR and FAR at that point are equal or the value is same at that point is termed as EER (Figure 25c). The EER is a

process can be used to give threshold independent performance measures. The lower the EER, the better the systems performance will be. [19]

4.3 Detection cost function, Detection Error Trade-off and Log likelihood: According to the NIST evaluation plan SRE-16, a basic cost model in a primary metric is used to measure the speaker detection performance and is defined as a weighted sum of miss and false alarm error probabilities given by the equation:

$$C_{Det}\ (C_{Miss}, C_{FalseAlarm}, P_{target}) = C_{Miss} \times\ P_{Target} \times P_{Miss|Target} + C_{FalseAlarm} \times (1 - P_{Target}) \times P_{FalseAlarm|NonTarget} \quad \textbf{……..(40)}$$

where the parameters of cost function are C_{Miss} and $C_{FalseAlarm}$ and P_{Target} are defined to have the following values:

Table 1: SRE16 cost parameters

Parameter ID	C_{Miss}	$C_{FalseAlarm}$	P_{Target}
1	1	1	0.01
2	1	1	0.005

In the actual detection decision, for each hypothesis, the score will be required. This score should reflect the system's estimate of the probability. To know the probability that the target speakers' speech is present in the segment, the scores will be used to produce Detection Error Tradeoff, in-order to calculate how misses maybe tradeoff against false alarms. Since these curves includes all the trials in each test for all targeted speakers, it should consider the normalized vectors for scores. Thus, DET takes the list of prediction values and a list of true values, to produces an output containing FAR and FRR for all possible threshold values.

The DET curve (Figure 26), the line showing the tradeoff between FAR and FRR is typically viewed in a log-log plot. Figure 26 illustrate the DET curve, as it describes, FRR as y-axis and FAR as x-axis and a linear line which intersect the value is equal to the point at which FAR is equal to FRR and is the value of EER. The scores used for DET are more informative. Thus, it is suggested that participants provide as scores estimated log likelihood ratio values which do not depend on parameters. So non-target hypotheses of the log likelihood ratio are given by:

LR = prob (data | target hyp.) / prob (data | non-target hyp.) **……..(41)**

A log likelihood ratio (LLR) based cost function, which is not application specific and may be given an information theoretic interpretation, is defined as follows:

$$C_{LLR} = 1/(2 * log2) * (\sum \frac{\log\left(1+\frac{1}{s}\right)}{N_{TT}}) + (\sum \log(1+s)\ /N_{NT}) \text{ .....(42)}$$

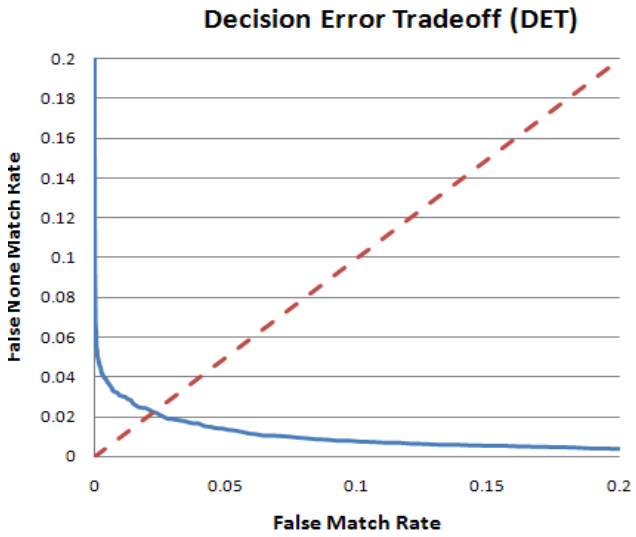


Figure 24: DET Plot

4.4 Receiver Operating Characteristics (ROC): These curves are used to depict the any predictive model which differs between the true positives and true negatives. Basically, roc curve does this by plotting sensitivity, the prediction of positive to positive, against 1-specificity, the probability of predicting negative to negative. This is how it shows accurately how accurately the model is predicting positives and negatives separately. Figure 27 is an example plot for this.

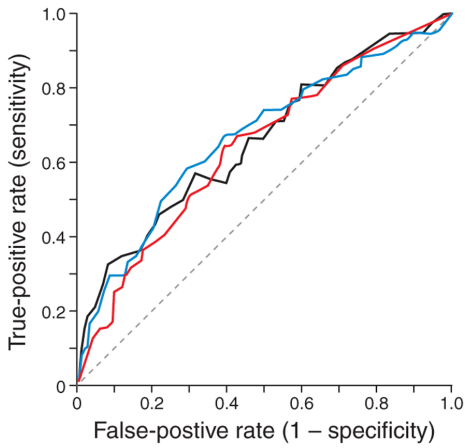


Figure 25: ROC example Plot

4.5 Scoring (True positive, True negative, False positive, False negative): By measuring whether the model assigns the correct class data to the test instances, one can evaluate the model's performance. For instance, consider a classification with 2 classes, positive and negative. Given a classifier and an instance, since there is only two possibilities, there will be 4outcomes.

True Positive (TP): If the signal is positive and it is classified as positive.

True Negative (TN): If the signal is negative and it is classified as negative.

False Positive (FP): If the signal is negative but it is classified as positive.

False Negative(FN): If the signal is positive but it is classified as negative.

4.6 Accuracy: This is the simplest scoring measure. It calculates the proportion of correctly classified instances given by: Accuracy = (TP+TN)/(TP+TN+FP+FN). ……**(43)**

4.7 Specificity: This is also called as True Negative Rate, which relates to the classifier's capability to identify negative results given by: Specificity = TN/(TN+FP). **……(44)**

4.8 Sensitivity: This is also referred as True Positive Rate, which is the proportion of actual positives correctly recognized as positives by the classifiers given by:

Sensitivity = TP / (TP + FN) **…….……(45)**

4.9 Precision: This is a measure of retrieved instances that are relevant.

Precision = TP / (TP+FP) **…….……(46)**

4.10 Recall: Recall describes the result which all labelled correctly. It is defined by the equation, Recall = TP/(TP+FN). When the system is very accurate, recall and sensitivity will be the same.

4.11 F-score/ F-measure: with the precision and recall, the harmonic mean of Precision and recall is referred as F-measure.

F1 = 2 ((P * R) / (P+R)) **……(47)**

5. ARCHITECTURES AND SPECIFICATIONS

As the tasks mentioned in aim and the methods which are explained briefly in previous sections, Comparison of Speaker Identification methodologies and Voice impairments detection are the main aim of the study. Since all the challenges have different implementations, this section illustrates the architectural block diagrams for the said purposes.

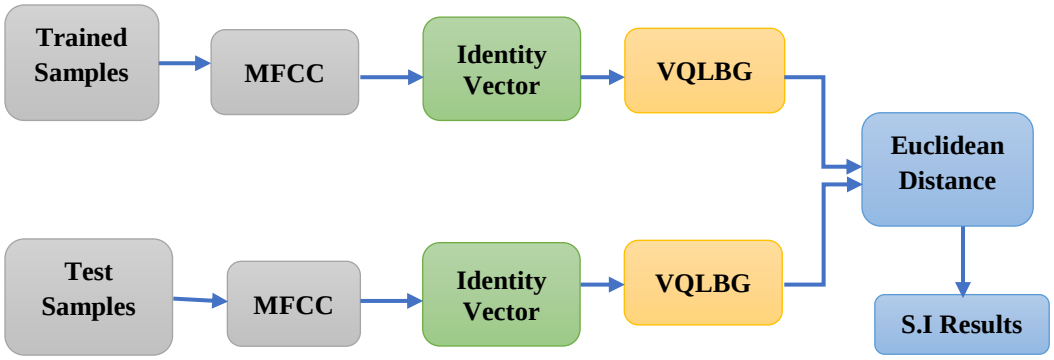


Figure 26: Block diagram of Speaker Identification using Euclidean Distance Scoring Method

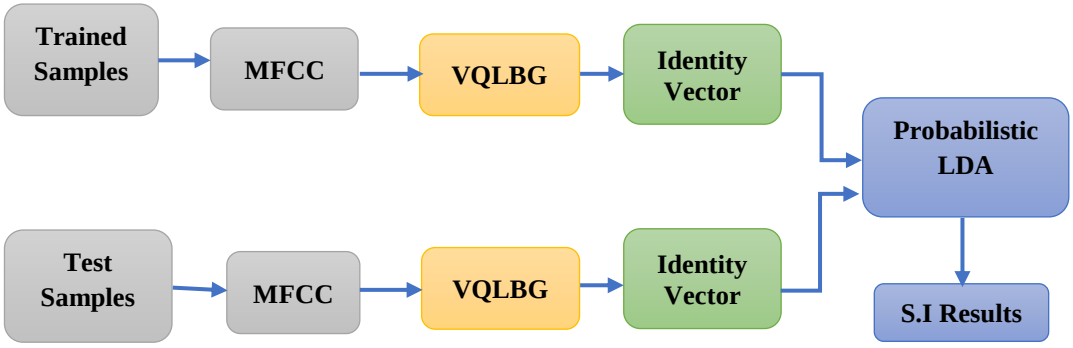


Figure 27: Block diagram of Speaker Identification using PLDA Scoring method

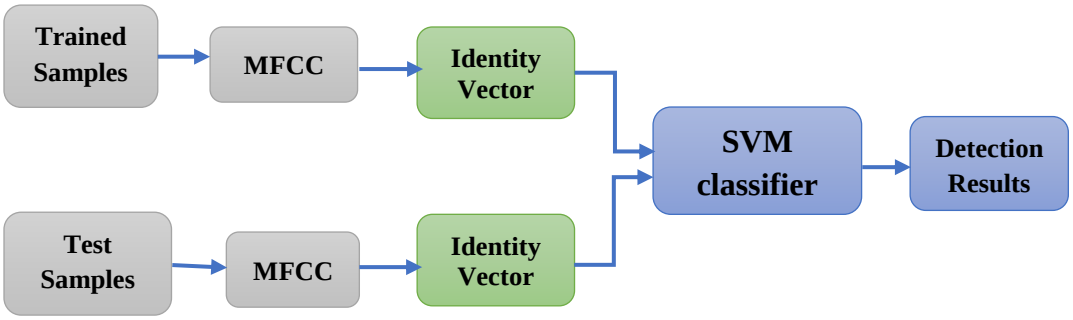


Figure 28: Block diagram of Voice impairments detection

Figure 28, 29 and 30 are the main architectures of the research work as they depict, feature extraction, classification and matching for the identification process and for voice

impairments detections and further in the procedures and results, comparison between Euclidean Distance and PLDA will be explained. Figure 31 illustrates a small implementation of real-time identification with Euclidean Distance scoring method and furthermore details will be explained in next sections.

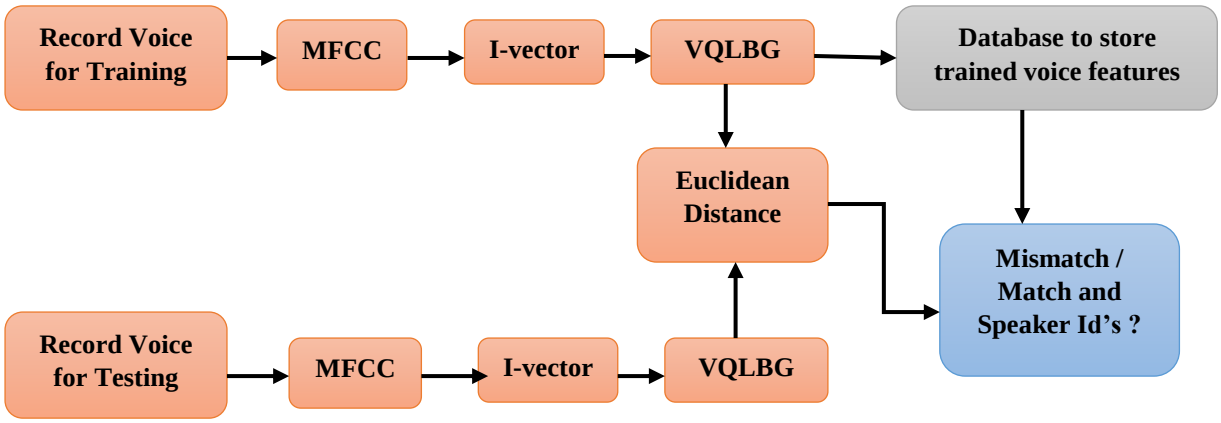


Figure 29: Block Diagram for Real-time identification

Specifications:

1. **Software Configuration:** MATLAB is the high-level language and interactive environment using by most of the engineers and scientists around the world. The matrix-based language is the best way to express computational mathematics. There are many advantages in the updated version of MATLAB R2018a along with other new updates, data analytics with statistics and machine learning toolbox, text analysis toolbox and predictive maintenance toolbox are updated. In this, SVM for Big data algorithm is implemented and also as MSR Identity Toolbox is used for some implements.

   MATLAB and Simulink - R2018a (64-bit)

2. **Computer Configuration:** This is mentioned because of real-time implementation for which System's Realtek high definition audio default device for playback and recording (Speakers and Microphone array) are used.

   Windows 10, Intel(R) Core(TM) i5 – 7200U CPU @2.50GHz, 2.71GHz, 64-bit Operating System x64 – based processor.

6. SPEAKER IDENTIFICATION AND COMPARISON OF METHODOLOGIES WITH RESULTS

From literature survey, one can find many possible methodologies for the Speaker Identification purpose and their results. To give the best methodologies by comparing the created new two methodologies will be explained in this section. Basically, mel frequency cepstral coefficients for feature extraction, vector quantization for feature classification and matching are observed, but in this study, two new methodologies which have different methods for all the steps of the separate feature techniques. To prove that first methodology or both may work for real time recognition, this small flow is implemented.

6.1 Real – Time Speaker Identification:

Firstly, to implement in real-time, Mel frequency cepstral coefficients for feature extraction, Identity vector and vector quantization as feature classification and Euclidean Distance as feature scoring techniques are applied. The brief working elaboration of all these methods are well explained in chapter 3 and Figure 31 shows the architectural block diagram of the concept.

The implementation or the process flow is portrayed in the Figure 32 i.e., flowchart of the complete process is emphasized. The MATLAB codes can be find in Electronic disk (CD) (Refer Appendix). As per this, there are six options in the Speaker Identification Application Menu Figure 33a. There are described as follows:

i. New database: To add voice samples to the database.
ii. Recognition Input: by loading a test signal to verify whether the speaker is same or not.
iii. Load sound for testing: To check the microphone, speaker and recording by loading a sample.
iv. Database information: Voice samples which are stored for training.
v. Delete Database: To delete the voice samples.
vi. Exit: to exit the menu.

The output displays and dialog boxes are shown in the figure 33.

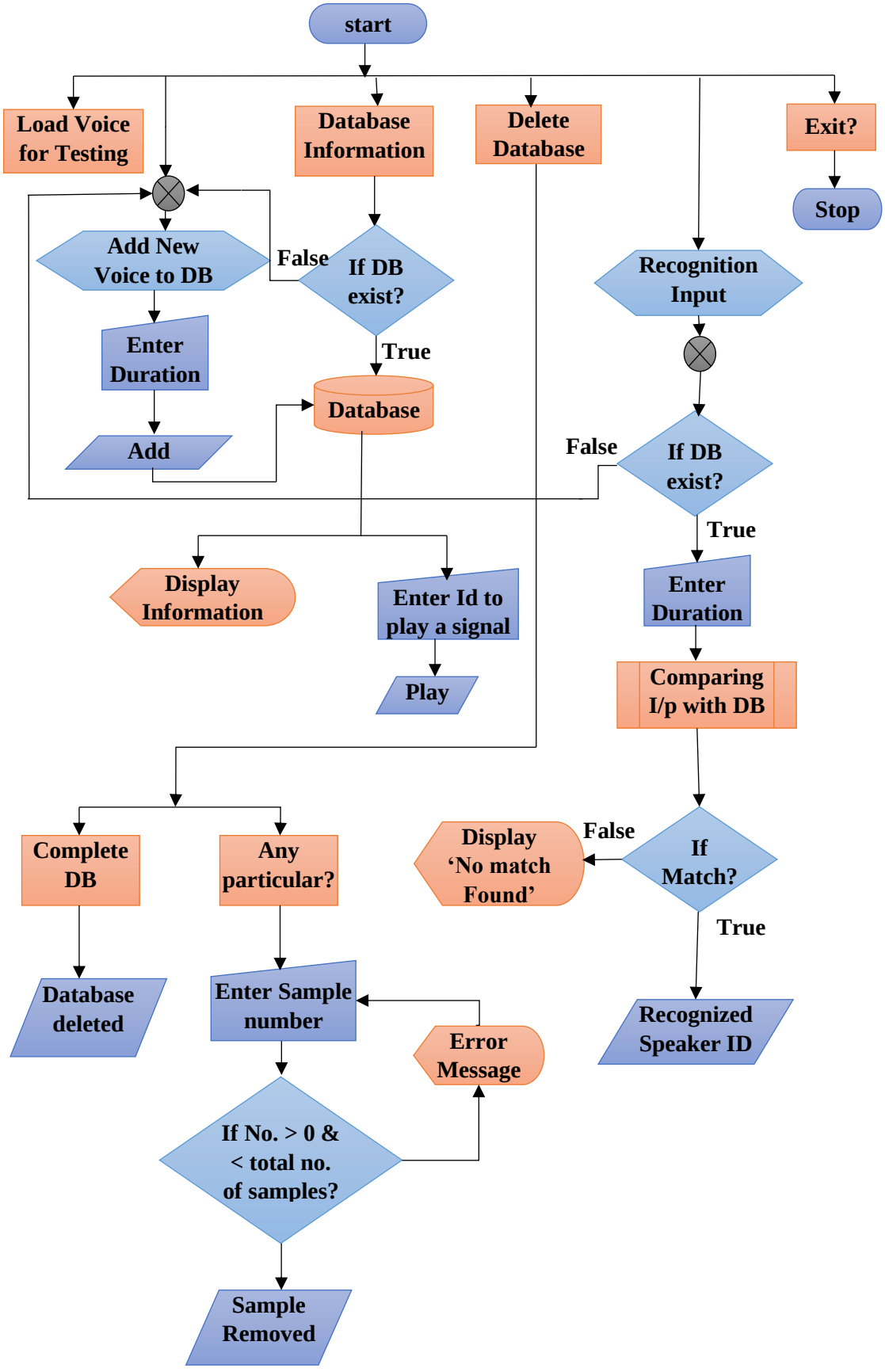
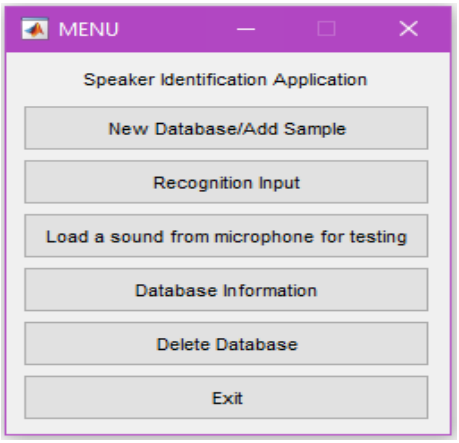
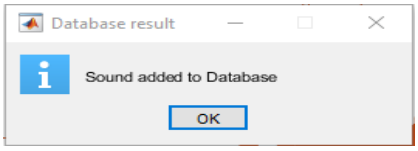


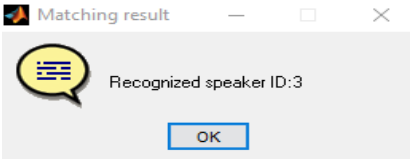
Figure 30: Flowchart of Real-time Speaker Identification



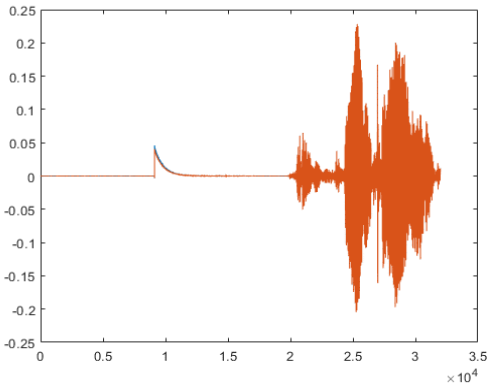
Dialogue a:Menu



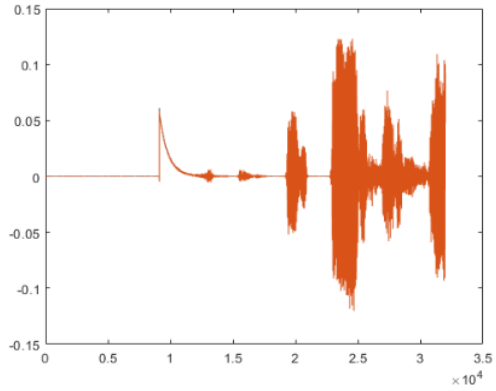
Dialogue b: Sound added to database



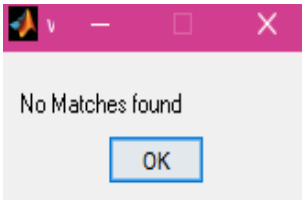
Dialogue c:Recognition dialog

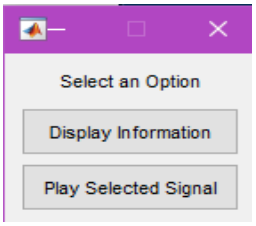


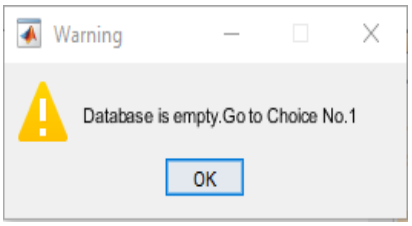
Dialogue d:Trained speaker signal plot

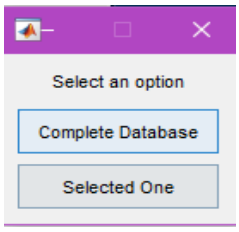


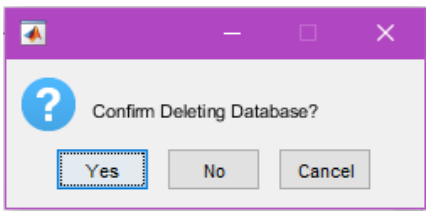
Dialogue e:Tested speaker Signal plot (Recognized)

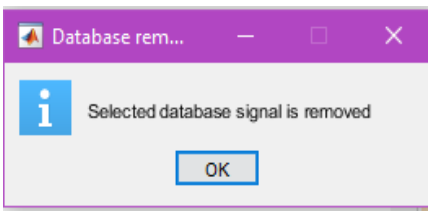


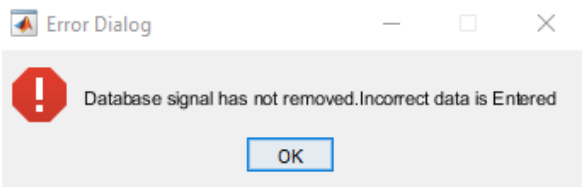


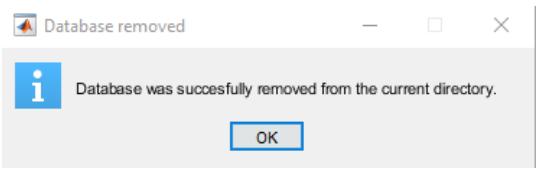












Dialogue f: Multiple Dialogues for different options as per the flowchart

Figure 31: Output displays and dialogues for real-time identification

Thus, overall from the above demonstration, it is clear that Speaker Identification in real-time with all the methods mentioned is possible but not very accurate though. Hence, to measure the performance, the same method is implemented with different databases in the next section.

6.2 Speaker Identification with Euclidean Distance: For the text independent and for multilingual speaker identification, the methods which mentioned above are implemented (figure 28). To prove the multilingual conditions, Uyghur and Korean Languages are taken along with Libri Speech (English) databases.

6.2.1 Databases Information:

1. Libri Speech (English): Libri Speech is a corpus of approximately 1000 hours of 16k Hz read English speech, prepared by Vassil Panayotov with the assistance of Daniel povey. The data is derived from read audiobooks from LibriVox project and has been carefully segmented and aligned[21]. Its purpose is to enable the training and testing of Speech recognition.

   The corpus is divided into several parts to enable users for the selective download and hence from this big database, dev-Clean.tar.gz has taken for the further process. Out of 40 speakers, 30 samples for each 30 speakers are taken for training for the process and 10 samples for each of the same 30 speakers are for the testing for the FRR, and 5 samples for each of different 10 speakers are taken for testing for the FAR. The average duration of maximum samples is between 5 seconds to 15 seconds.
2. Uyghur Language (Turkic language): THUYG-20: THUGY20[22] is an open Uyghur speech database published by center for speech and language technology, at Tsinghua university. It involves the full set of speech and language resources required to establish a Uyghur speech recognition. This huge database is split into few parts, in that one-part data pack is taken for the purpose. This database consists of enrolled 80 speakers, 40 Male and 40 Female speakers duration of between 3 seconds to 20 seconds averagely. For training 15 signals per person from 40 female speakers and 40 male speakers, for testing as proceedings for FRR, 3 signals per person from the same 20-20 speakers and 3 signals per person from 10 different speakers as 5 male and 5 females. Thus, total enrolled speakers are 90.
3. Korean Language: This is an Open-source speech corpus for speech recognition by Zeroth project. The dataset contains transcribed audio data for Korean [23] and it is created to fulfill the purpose of automatic speaker recognition from Korean language. Thus, from this big database, in-total 60 enrolled speakers are taken in which 40

speakers for training and 20 speakers for training(Impostors). To explain in detail, 15 signals per person for 40 speakers are taken for training, the same speakers 5 signals per person is taken for testing as FRR and 20 speakers 5 signal for each is taken for testing as FAR.

Hence, by considering all these languages, context will be proved for language independent methodologies, though they give slight errors for accuracies but will be mostly successful.

Now, to examine the speaker identification part, the architectural block diagram which represents this methodology is depicted in figure 28 which follows the proceeding methods: MFCC, I-vector, Vector quantization and Euclidean Distance. The MATLAB codes can be find in Electronic disk (CD) (Refer Appendix). Now the process flowchart and Experimental results will be discussed further.

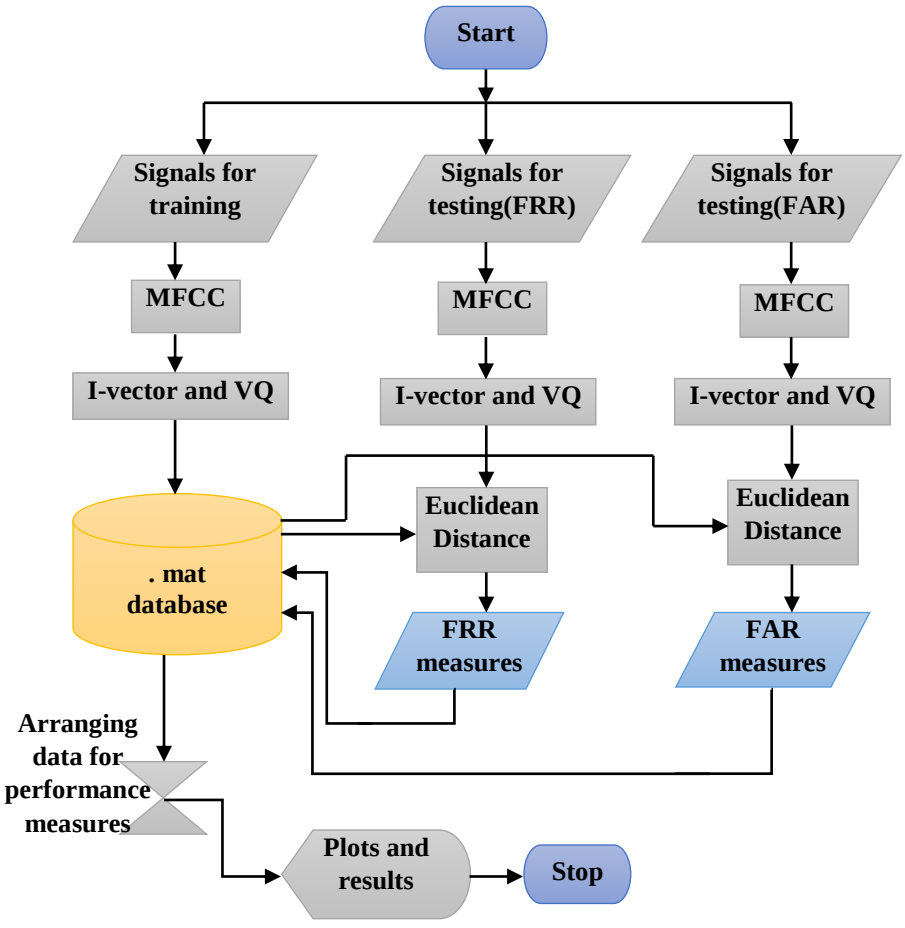


Figure 32: Flowchart for Speaker Identification (Euclidean Distance as scoring)

Above flow chart illustrates how the flow of the process for speaker identification is undertaken steps by steps, using Euclidean Distance scoring method.

6.2.2 Experimental Results:

1. Experimental Results for Libri Speech Database: Figure 35a presents the Detection error tradeoff curve, which is the output of the thresholding by Euclidean Distance and is plotted for every trade-off points. As it clear from the plot that, using all the mentioned methods and for Libri Speech under the said conditions it is valuable to note the EER of 10.2941% which is very reasonable. Figure 35b, represents the FAR, FRR and TSR curves which in results proves the EER. Receiver characteristics curve is also shown in figure 35c, which represents the better characteristics of the system performance. Further minimum Detection cost function according to the NIST specified parameters, at P_{target} = 0.01 is 0.1400 and P_{target} = 0.005 is 0.0700 which is also best minimum value one can get using Euclidean Distance classifier.

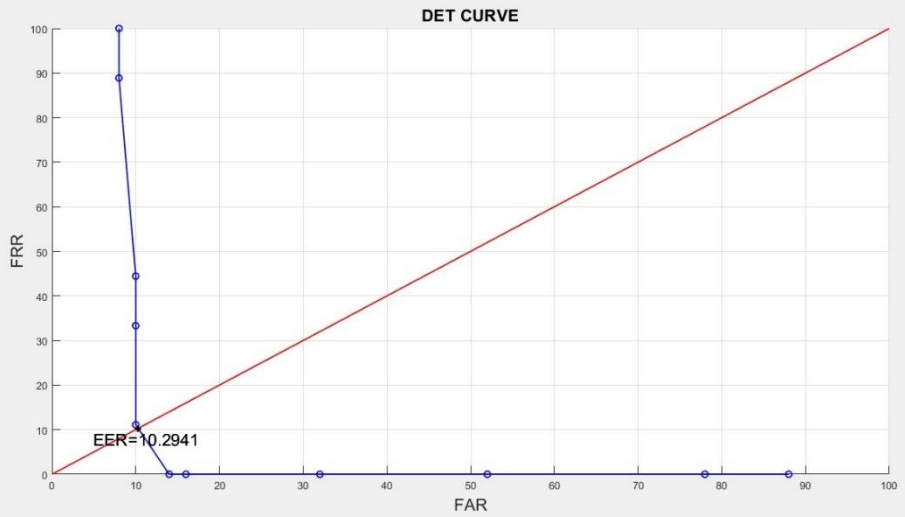


Figure 33: a) Detection Error Tradeoff curve for Libri Speech Database

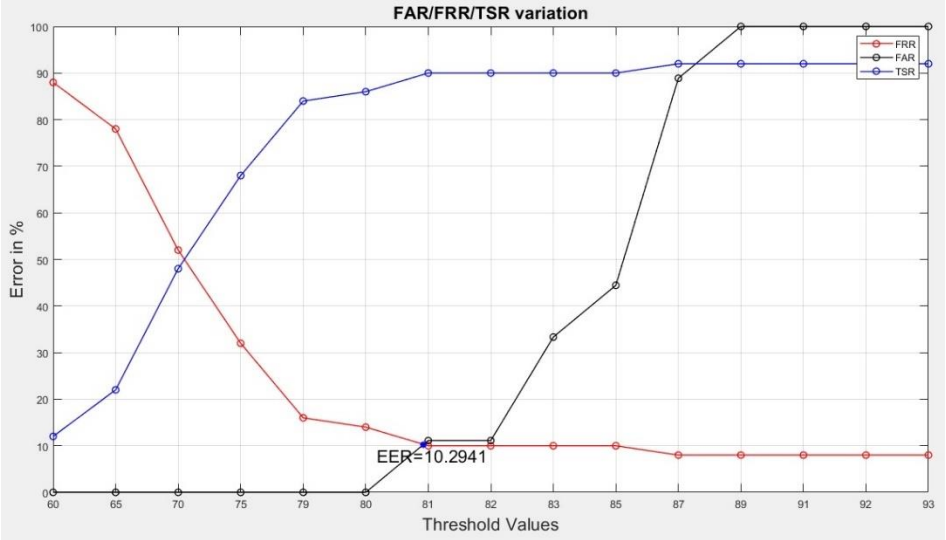


Figure 35: b) Thresholding FAR, FRR, EER and TSR plots

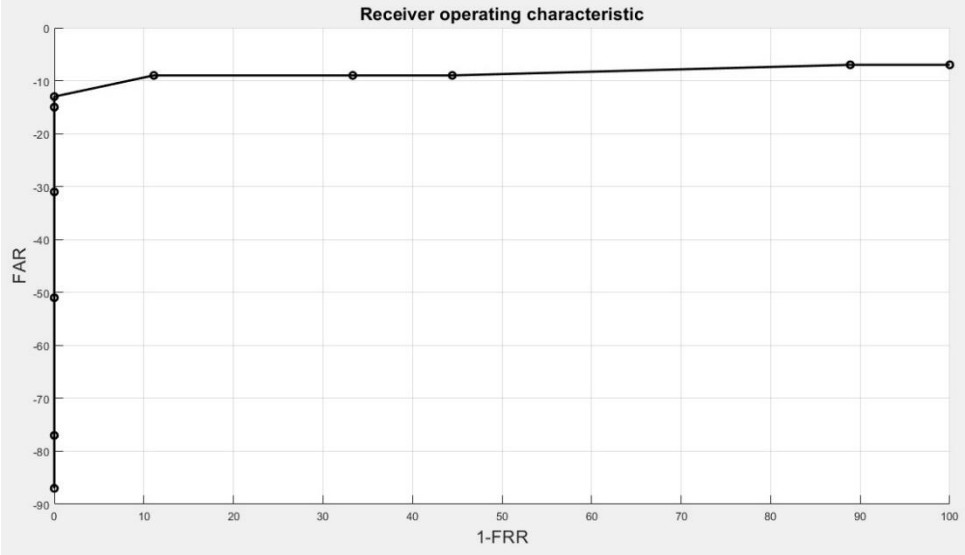


Figure 35: c) ROC of Libri Speech database

2. Experimental Results for Uyghur Language database: Figure 36a shows the DET curve of the performance, here one can observe, how language difference is impacted on the results of the identification. And from the plot, with the help of linear line plotting, EER is 23.1884%. The FAR and FRR curves also proves the EER (Figure 36b), also TSR is at its best and showing the best performance of the system. Equal error rate is also obtained from the algorithm and displayed in the command window in MATLAB which is exactly the same number displayed in the plots. ROC is also depicting the characteristics of the system by the plot from the figure 36c. Additionally, as the minimum Detection cost function obtained by the algorithm, according to the parameters specified by NIST, at P_{target} = 0.01 is 0.6167 and P_{target} = 0.005 is 0.3083. Compared to the previous result from Libri Speech, it is increased. One more important note is that thresholding also depends on the database and its characteristics, for example like duration, recorded frequency, noise ratio etc.,
3. Experimental Results for Korean language database: Figure 37a illuminates the Detection error tradeoff curve for this database. By comparing previous two DET curves, one can find the difference about the system performance and its changing on fact of thresholding and it is proved in the given plot. Since the area under the curve represents the characteristics of the system, by the obtained plot, it is on average when compared to previous DET curves. It might be also depending on the other factor and parameters which mentioned before. Thus, EER for the given system is 20% by both DET plot Thresholding plots (Figure 37b) and ROC curve is represented by figure 37c which also depicts the

system characteristics by its operating points. The curve has also continuous points where it shows the error stability instead of increasing improvement when compared to the previous DET curves. However, according to the TSR, even with the obtained error rate, system's performance is at the best. Total success rate is also important to note because it shows the success rate regardless of the errors. To add to this, minimum DCF which is obtained according to the parameters specified by NIST, at P_{target} = 0.01 is 0.6222 and P_{target} = 0.005 is 0.3111. Minimum DCF values are also almost as same as the previous language database where it shows the decreased result when compared to Libri Speech Database.

Conclusion: Thus, to conclude the methodology which is implemented using mentioned methods and flow, Libri Speech database is more stable for good results and performances compared to other two languages, though they also showed reasonable results.

Usually the experiments which are enumerated in literature survey, which mentioned proved result with small databases. But the experimental result which are mentioned in this research are predominantly big datasets compared to them in the other researches. If one can consider very reasonable small database to obtain results, these methodologies also can be resulted in very desirable results. Although the reason behind considering for the big databases is that to prove the algorithm works better for big databases and even with different languages. However, in next section these results are also compared with other methodology.

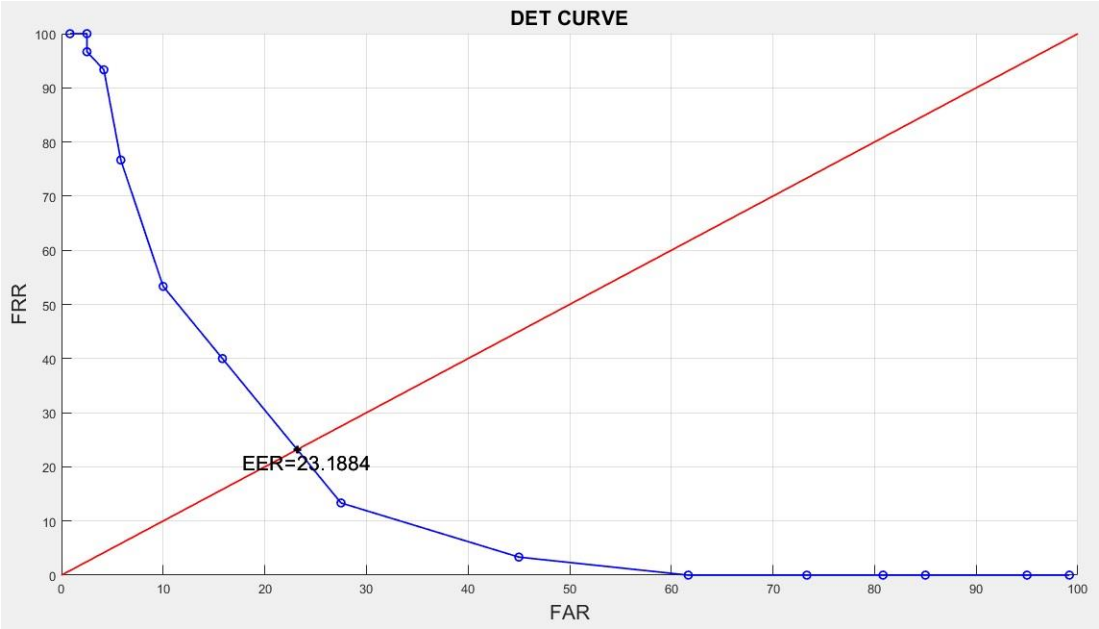


Figure 34: a) Detection Error Tradeoff curve for Uyghur speech Database

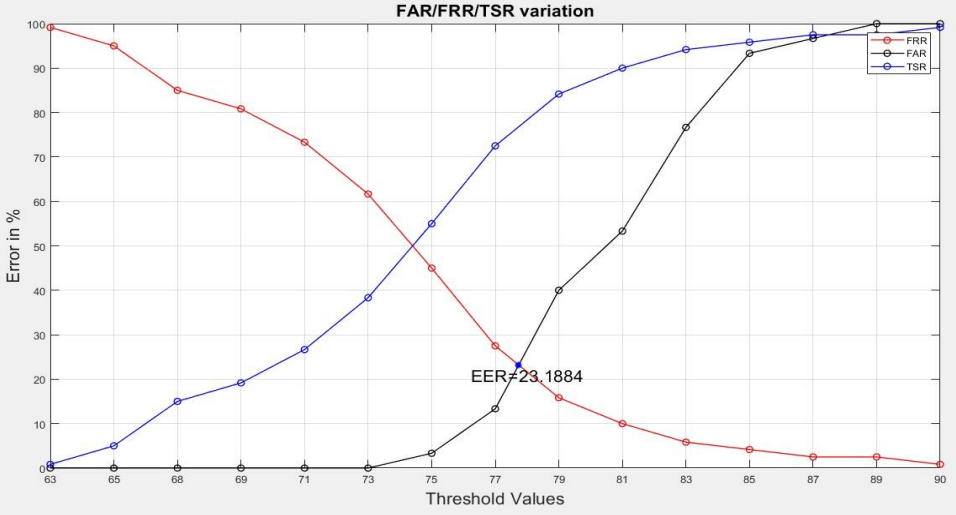


Figure 36: b) Thresholding FAR, FRR, EER and TSR plots

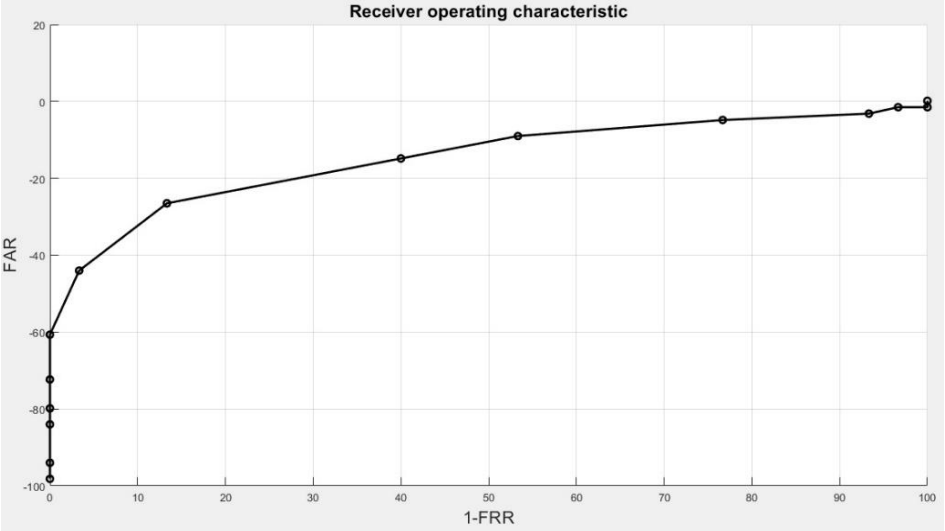


Figure 36: c) ROC of Uyghur Speech database

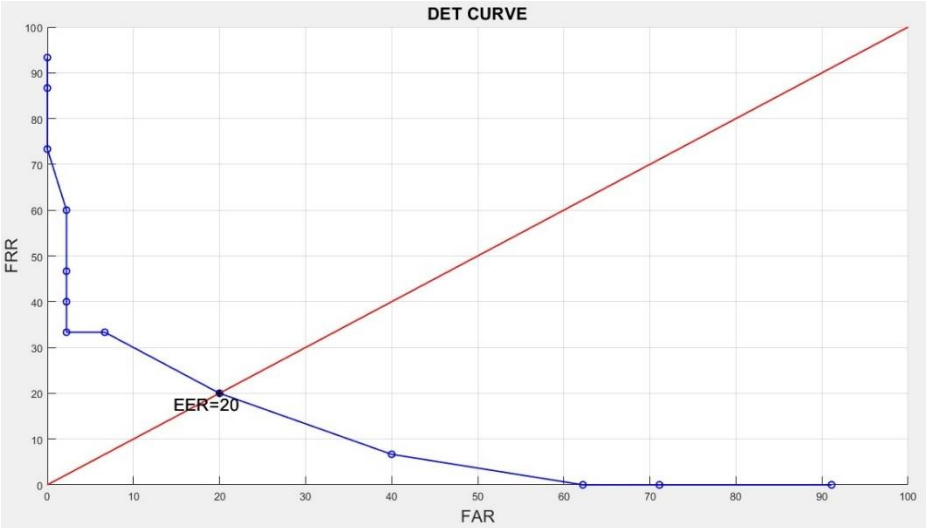


Figure 35: a) Detection Error Tradeoff curve for Korean Language speech Database

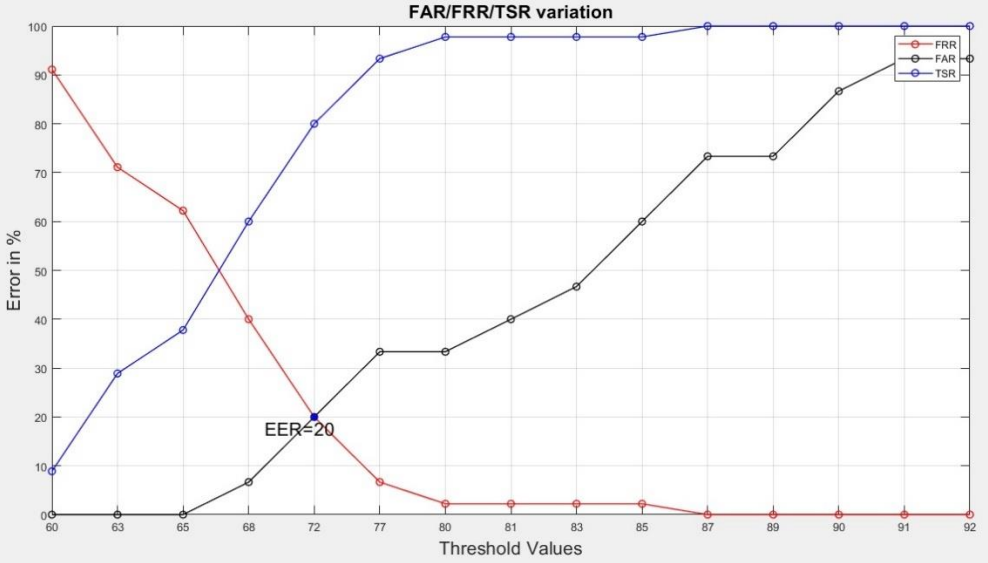


Figure 37: b) Thresholding FAR, FRR, EER and TSR plots

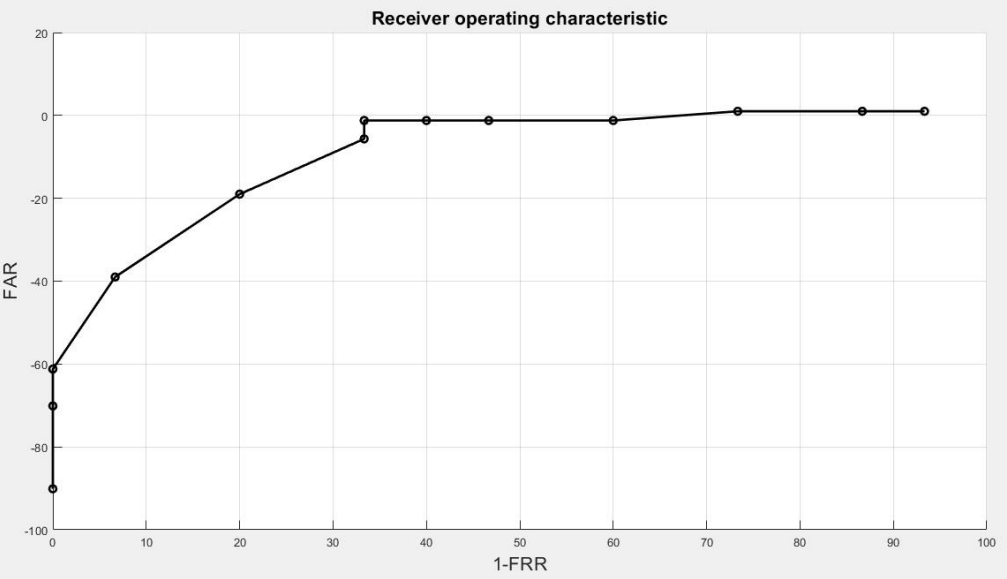


Figure 37: c) ROC of Korean language Speech database

6.3 <u>Speaker Identification with PLDA:</u> As like the implementation with the Euclidean Distance classifier, Speaker identification with probabilistic linear discriminant analysis scoring method, this section is lied. Here also text and language independent speaker identification will be undertaken. The MATLAB codes can be find in Electronic disk (CD) (Refer Appendix). Nonetheless the implementation flow will be little different which is as shown in the figure 29 and figure 38 is the process flow chart of this methodology based on how it works. Feature extraction will be MFCC, Feature classification will be Vector quantization and then i-vector and for PLDA as the feature matching techniques are experimented. The reason behind the alteration of the classifier is that PLDA takes the

vector input for the matching process to reduce the run time of the algorithm. Further comparison details will be explained in next section. In PLDA algorithm, before giving the input vector to the main algorithm, it has to be metered with some mechanisms, in such a way that output from the i-vectors has to reduce dimensionality by LDA and then it should be centred by mean, then applying the whitening transformation for the training set alone and then again centring and whiten all the i-vectors to projected into a unit sphere. Then Average enrolment data is taken and after sorting PLDA variants will be applied. As explained in Chapter 3, PLDA has 3 types of variants and all the three variants are implemented although only one can choose at a time and according to it, it will compute the results. Further, according to the performance measures in chapter 3, EER, min DCF, DET and Log likelihood are calculated based on how the variants are specified.

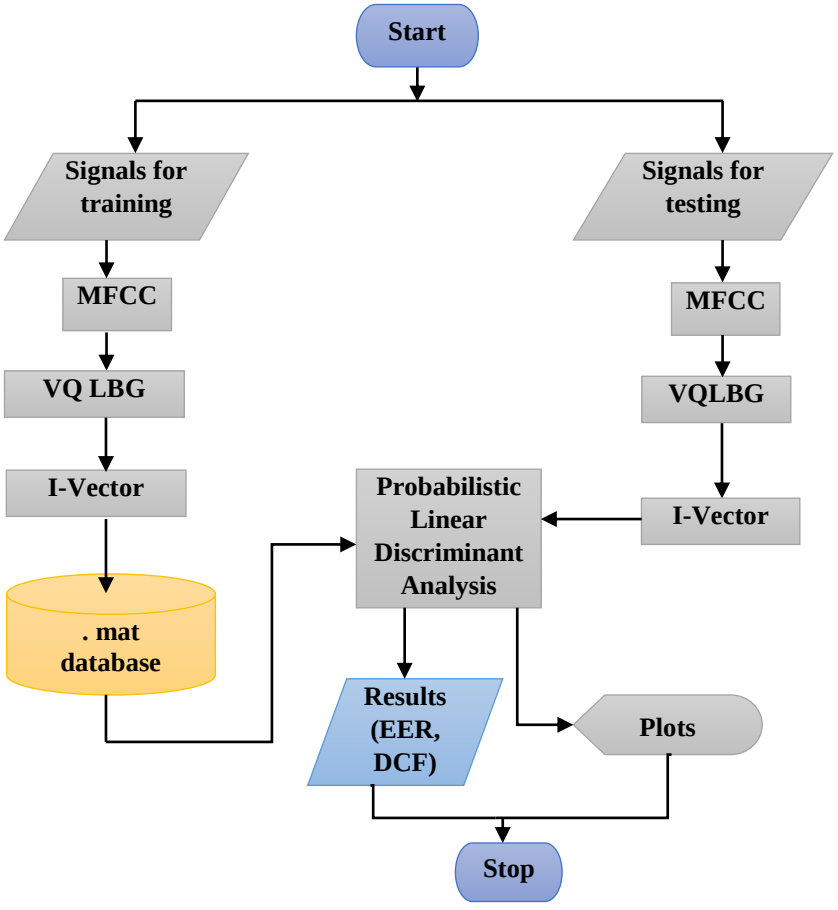


Figure 36: Flow chart of speaker identification with PLDA as scoring method

6.3.1 Experimental Results:

1. Experiment results for Libri Speech Database: The Database samples are considered the same for both the classifications due to the consequence for comparison. Figure 39 demonstrate the Detection error tradeoff curves for all the trade-off points. Since PLDA

takes every samples clarification and the data has been metered, the curves will be smooth and hence 15 iterations are taken to choose the best by repeating process, it also has 15 DET curves which appear like one curve indeed, all differ from very decimal points. Nonetheless the plot is reduced to 10x10 since data is metered and to be very precise. Correspondingly, by the plot EER also has 15 values which differ in the 4^{th} or 5^{th} decimal numbers. Additionally, LDA's different dimensions are also considered according to the best values for the dataset. Aforementioned resemblance for P_{target} values, for the best minimum DCF's which is obtained according to the parameters specified by NIST will be at $P_{target} = 0.01$ and $P_{target} = 0.005$. The Table 2 decipher the results, the EER and the DCF values obtained are the minimised results from many iterations and the lowest one is recorded. Now, as per the minimum and best EER result from the table, for those values, DET is also plotted and shown in Figure 39.

Table 2: Speaker Identification Results for Libri Speech.

	LDA Dimensions (< 40)					
	14		16		18	
P_target values	EER	DCF	EER	DCF	EER	DCF
0.01	4.8256	0.0142	4.1554	0.0383	4.7181	0.0469
0.005	4.8256	0.0092	4.1554	0.0335	4.7181	0.0421

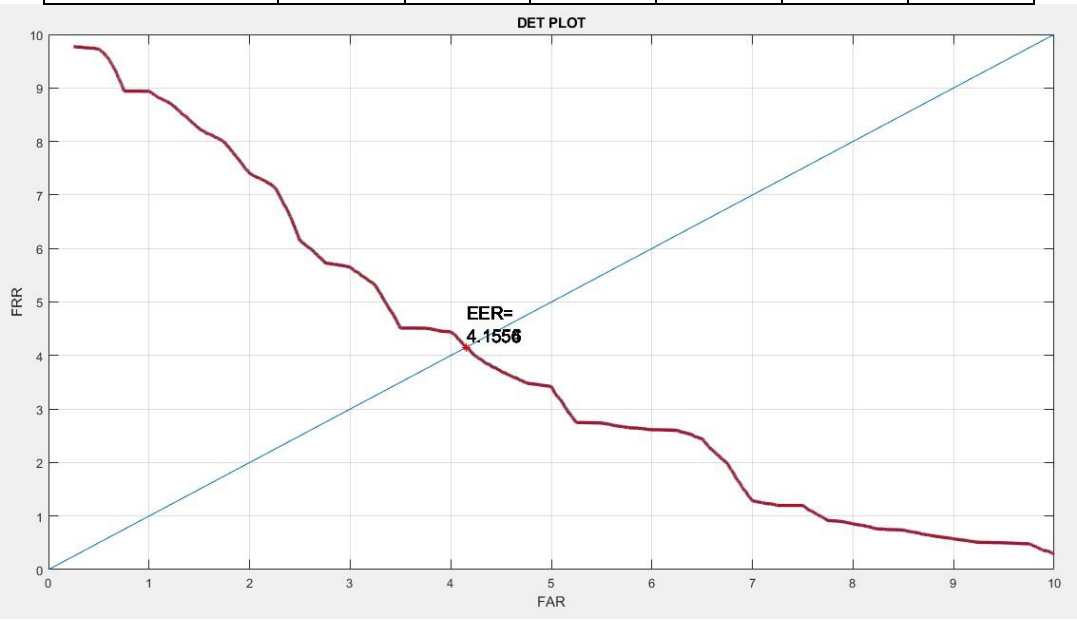


Figure 37: DET plot for Libri Speech Database.

2. Experiment results for Uyghur language Database: For this explication also, same like Libri Speech implementation and procedures. But the LDA dimension is differ from the one before regardless on the case of the database and enrolled speakers. Though the LDA dimension vary on the basis of the database, thereon LDA dimension should be greater than zero and less than the number of individual speakers or the enrolled speakers. Figure 40 depicts the DET curve for the considered database and Table 3 illustrates the results with the different P_target values. Here 10,16 and 64 LDA dimensions are considered, since they have the best optimal values for the considered database.

Table 3 : Speaker Identification results for Uyghur Database

	LDA Dimensions (< 100)					
	10		16		64	
P_target values	EER	DCF	EER	DCF	EER	DCF
0.01	4.4742	0.0128	4.5023	0.0180	4.3959	0.0102
0.005	4.4742	0.0078	4.5023	0.0131	4.3959	0.0052

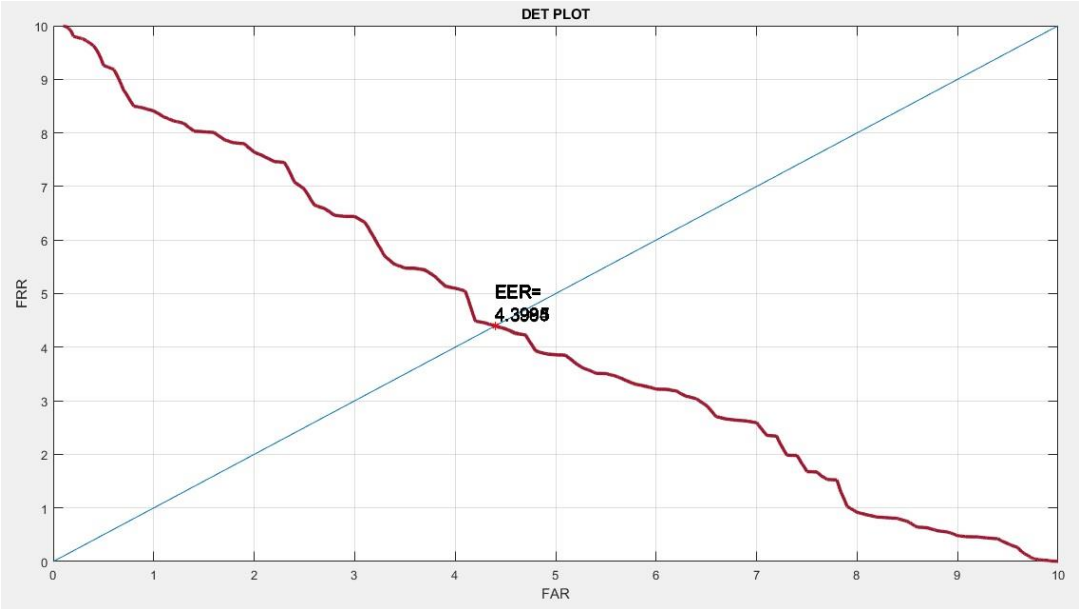


Figure 38: DET Plot for Uyghur Database

3. Experiment results for Korean language Database: Korean Language database which is as mentioned before in the above section, it is also considered how it was computed for Euclidean Distance. Table 4 describes the outcome of this databases consequences. And the figure 41a exemplify the DET curves for the outcome. EER and minimum DCF's are presented with multiple possibilities.

Table 4:Speaker Identification Results for Korean Database

	LDA Dimensions (< 60)					
	20		24		37	
P_target values	EER	DCF	EER	DCF	EER	DCF
0.01	4.3820	0.0144	4.4082	0.0220	4.1801	0.0182
0.005	4.4742	0.0094	4.5023	0.0171	4.3959	0.0132

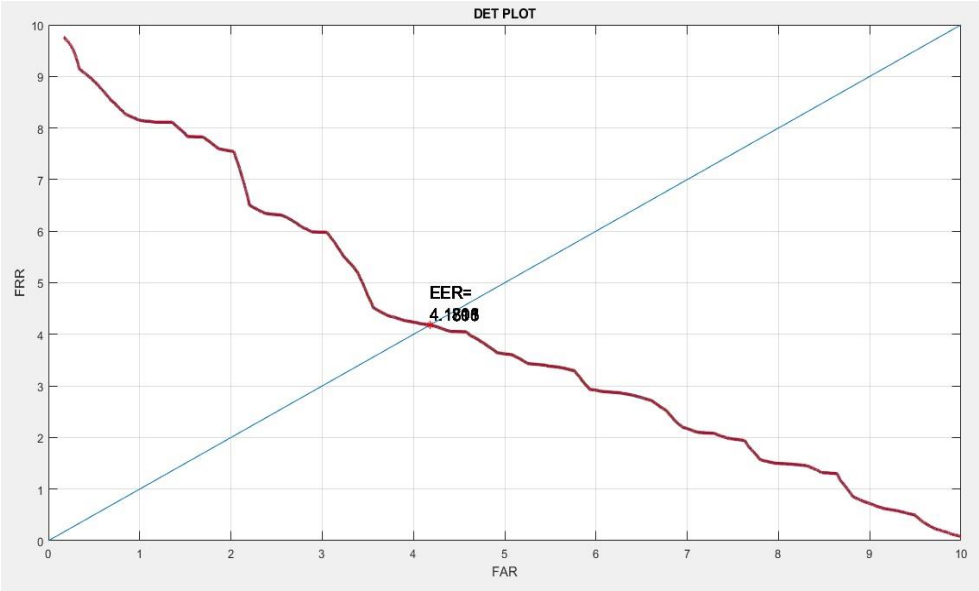


Figure 39: a) DET Plot for Korean Database

Conclusion: Thereupon, to summarize, Libri database with clean speech gave the best outcome when compared to other two databases. Though the results are threshold independent in PLDA, it depends on other factors like recorded Frequency, records duration etc., so clearly analysis and arranging all the possible parameters may give better results.

6.4 Comparison of Euclidean Distance and PLDA after results: As one of the main goal of this research, comparison between the euclidean distance and PLDA as the scoring methods is with good turnout. It is evidently and very certainly visible that PLDA scoring method is the best with result, by handing best EER and DCT outputs with DET curves. In the section 6.2 and 6.3, different experiments done for different databases using these two scoring methods. Difference between databases are explained in their corresponding. Hence, to conclude, Libri Speech Database gave the best outcome, in-terms of language

database and in term between Euclidean Distance and PLDA by bestowing EER and DCF with DET plots. However, other two language databases also guaranteed the unsurpassed result to compare with other researches as in literature survey. Thus, all languages databases results are perfectible and idyllic for the Speaker Identification process.

7. VOICE IMPAIRMENTS DETECTION

The voice impairments detection in this research is contrasting to the researchers from [6][7][8] by whipping the results backward. The experiments with LBP which compared with MFCC concluded that instead of MFCC, LBP is better for pathological voice detection. Similarly, another experiment with MFCC and SVM mentioned that pathological voice detection gives 93% accuracy. By considering all these, the study here is explained and experimented. From Chapter 5, figure 30 depicts the processing block diagram of the voice impairments detection and further processing steps will be explained here.

As per the block diagram: figure 30, MFCC is carried out for features extraction, i-vector for features classification and SVM for feature matching. And then the sorting of the data from the SVM is the output of this methodology where it decides whether the sample is normal or abnormal and then will exhibit the performance measures. Feature process methods are explained theoretically in chapter 3. For the experiment, the TORGO database is considered and details are as follows:

7.1 Database Information: The TORGO database of dysarthric articulation consists of aligned and measures features from speakers with either cerebral palsy or amyotrophic lateral sclerosis which are the two most causes for speech disability. This database called TORGO, is the output of the collaboration between the departments speech language pathology and computer science[24]. The speakers are having some are CP and some are ALS resulted in dysarthria which is caused by disruptions in the neural motor commands to the vocal folds or articulators resulting in unintelligible speech. The main purpose of this detection is that, the dysarthric speeches cannot be recognized by Automatic speaker recognition being the reason that of the unintelligible speech data. However, SVM is the best for the classification in this case.

The Abnormal voice database which is used for the experiment is extracted from this big dataset, 3 female and 5 male abnormal speakers. From each 50 signals are extracted to create abnormal database for this experiment. For normal speaker's database, Libri Speech database's 350 signals are taken and for test both are mixed in 1.5:2 ratio. That is 150 normal samples and 200 abnormal samples are mixed and fed as testing database.

7.2 Experimental Results: By considering the above database and explained methodology, the following results are obtained. The MATLAb codes can be find in Electronic disk (CD)

(Refer Appendix). Simultaneously, figure 42 represents the process flow chart of the methodology in which it depicts how the data is processing and testing and matching.

The table 5 illustrate the total results of the Voice disorder detection in which all the performance measures are tabulated. As it is evidently visible that the Accuracy is 100%, and specificity and sensitivity also 100% depicts the 100% working rate of the methodology.

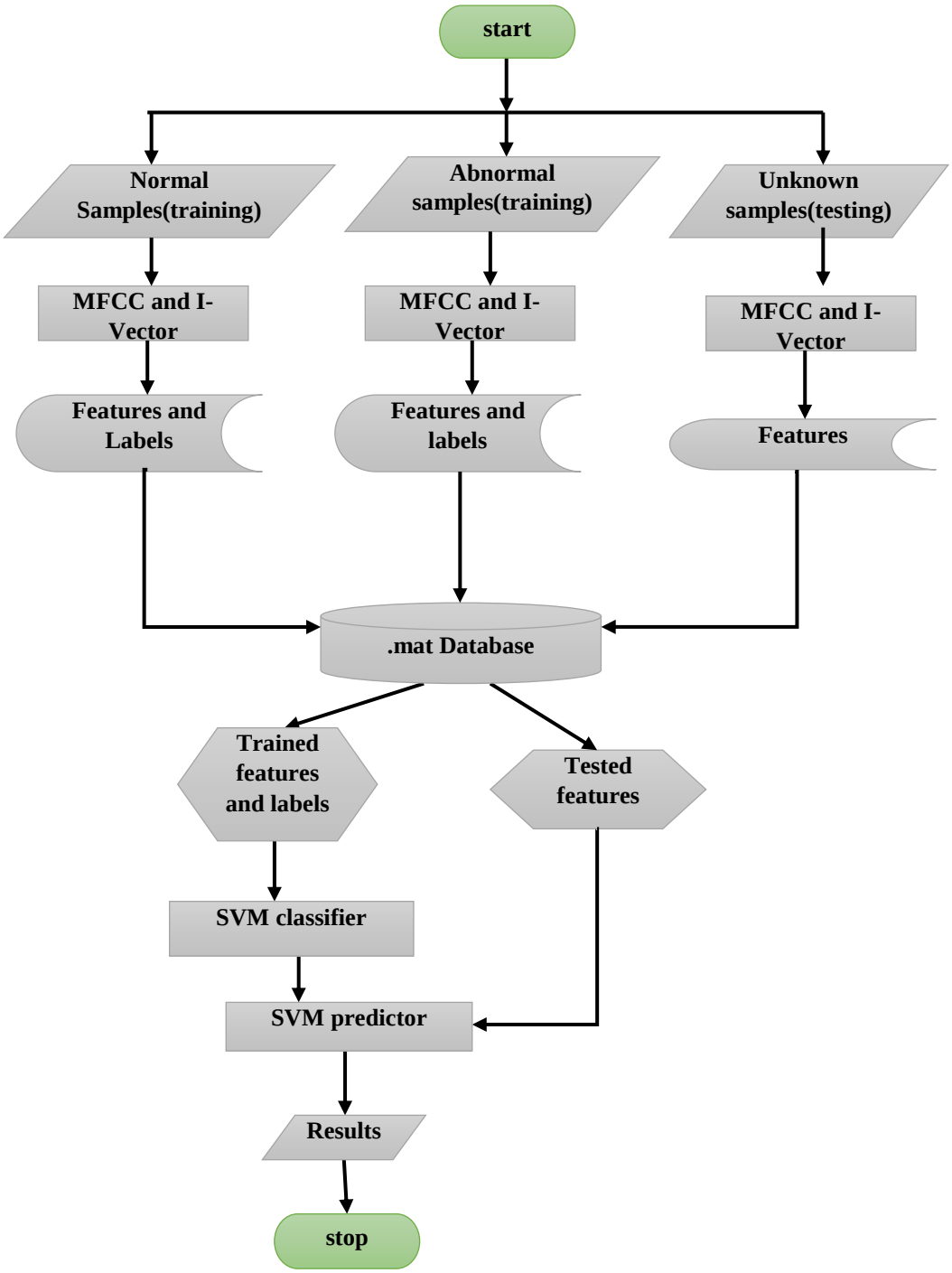


Figure 40: Flowchart of Voice disorder detection process

Table 5 : Basic Performance measures

Performance Measures	Scores in Numbers
True Positive	500
True Negative	550
False positive	0
False negative	0

Table 5: Performances measures for Voice impairments detection

Performance Measures	Scores (%)
Accuracy	100
Specificity	100
Sensitivity	100
Precision	100
Recall	100
F Measure (F score)	100

The idea behind it is very simple that algorithm which predict the labels of the classified classes are arranged properly by the SVM algorithm which is very precise and accurate. And as a result, precision and recall is also 100%. SVM classifier is a binary classifier which code-predict one versus one method. The true positives and true negatives showing the values of the samples which are detected by algorithm, as the definition of them depicts, it estimates the truly calculated positive and truly calculated negative samples numbers. Since it is the clearly saying no false positive and no false negative are found, obviously there can't be error in the result to figure out the negatives. Thus, it proves this is the best methodology to detect the dysarthric pathological voices over healthy ones.

8. CONCLUSION

The text and language independent speaker identification system is successfully implemented with two different new methodologies. By comparing multiple possibilities, such as classification with Euclidean Distance scoring methods using 3 languages databases, comparison between them profitably depicts that the use of Libri Speech is more effective when compared to the other two language databases with the terms of EER, DET and DCF values. And then comparing all the databases results with the other classifier, i.e., PLDA, the probabilistic linear discriminant analysis method is another scoring method which is compared against the Euclidean Distance. To forth see, here also, all 3 databases were under the experiment and evidently proved that the use of PLDA as scoring method brought the best results, by reducing EER from Euclidean Distance almost ranges from 10% to 20% to the range from 4.15% to 4.50% and minimum DCF values ranges from 0.5 to 0.6 to ranges from 0.01 to 0.005. Additionally, showing DET curves and other performance measures like ROC and TSR are also validated for the obtained results. However, other two language databases also guaranteed the unsurpassed result with the EER, minimum DCF and DET values.

And also, it is important to note that the PLDA prevail over the Euclidean Distance in some efforts like considered big database took 10 to 15 minutes to execute the program through algorithm in Euclidian distance whereas PLDA took just 3 to 5 seconds and also, this run time differs on the size of database and also on the system's processors speed. In this context, Windows 10, Intel(R) Core(TM) i5 – 7200U CPU @2.50GHz, 2.71GHz, 64-bit Operating System is used. Thus, run time/execution time of the PLDA is best when compared to Euclidean Distance. Additionally, since the big databases are used, thresholding values are also altered according to that and by using limited and same small sized databases in all the language may give better results than here. But it this context, it is not included, because of the aim, that to compare the performance measures to assure the best result by all means. Thus, with the big databases and trained sets the results are highly valid. Another important point is Euclidean Distance depends on Threshold values and PLDA is threshold independent for classification / Selection

As mentioned in the objective, to build an application in the MATLAB to identify the speaker in real-time is also implemented and documented with the results. The process flowchart which explains the algorithm flow and the obtained dialog boxes with the result dialog and plots are also mentioned.

Eventually, as the voice disorder detection also consumed the same methods which are used for speaker identification but with the different unique classifier/ feature matching technique. By considering the TORGO database which is an acoustic and articulatory dysarthric speeches database, it is proved with 100% accuracy that the algorithm is eligible to detect the disordered /abnormal voice successfully. The other performance measures which are tabulated in the corresponding section, shows that it has zero false positives and false negatives which results in obtaining the recall, precision and even F score as 100% which is the best result compared all of the researched methods. The only drawback which SVM has is that it also takes minutes together time as execution time when running through the algorithm, however it can be neglected if the database size is small or normal. But in order to execute for the big database, more classified vectors may help to run faster with this algorithm as like PLDA.

9. REFERENCES

[1]Martinez Jorge and Suzuki Masahisa Mabo," Speaker recognition using mel frequency cepstral coefficients (MFCC) and vector quantization (VQ) techniques", *in Proceeding of IEEE 22nd International Conference on Electronics and Communications and Computing - CONIELECOMP*, 2012, pp. 248-251.

[2]M. Senoussaoui, P. Kenny, N. Dehak and P. Dumouchel. An i-vector extractor suitable for speaker recognition with both microphone and telephone speech, *Journal of Information sciences and Technology*. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.180.3255

[3] Y. Jiang, K.A. Lee and L. Wang. (2014). PLDA in the i-supervector space for text-independent speaker verification, *Journal on Audio, Speech and Music Processing*, [Online-ISSN: 1687-4722]. Available: https://doi.org/10.1186/s13636-014-0029-2

[4]R. Hasan, M. Jamil, G. Rabbani and S. Rahman, "Speaker identification using mel frequency cepstral coefficients", *in Proceeding of ICECE 3rd International conference on electrical and computer engineering*, 2004, ISBN 984-32-1804-4, pp. 565-568.

[5]A.S. Thakur and N. Sahayam, "Speech recognition using Euclidean Distance", *International Journal of Engineering Technology and Advanced Engineering*, ISSN: 2250-2459, ISO 9001:2008 Certified journal, vol. 3, Issue 3, pp. 587-590, March 2013.

[6]M. Alsulaiman and K. Almutib, "Vocal fold disorder detection by applying LBP operator on dysphonic speech signal", *Journal in recent advances in intelligent control, modelling and simulation*. ISBN: 978-960-474-365-0, pp. 222-228, 2014.

[7]Z. Ali and M. Alsulaiman, "Vocal fold disorder detection based on continuous speech by using MFCC and GMM", *in Proceedings of IEEE 7th GCC Conference and exhibition*, 2013, ISBN: 978-1-4799-0724-3, pp. 292-297.

[8]C. M. Vikram and K. Umarani," Pathological Voice Analysis to detect neurological disorders using MFCC and SVM", *International Journal of advanced electrical and electronics engineering*, ISSN(print): 2278-8948, vol.2, issue-4, pp. 87-91, 2013.

[9]A. Khosravani and M. M. Homayounpour, "A PLDA approach for language and text independent speaker recognition", *Journal in Computer Speech and Language-Sciencedirect*, Vol. 45, pp. 264-269, 2017.

[10] Net Industries. [Online]. Available: http://encyclopedia.jrank.org/articles/pages/6556/Biometric-Technologies.html

[11] N. Singh, R. A. Khan and R. Shree. (2012). Applications of Speaker recognition, *International Conference on Modelling, Optimization and computing (ICMOC), ISSN: 1322-3126* [Online]. Available: https://doi.org/10.1016/j.proeng.2012.06.363

[12] MedicineNet.com [Online]. Available: https://www.medicinenet.com/script/main/art.asp?articlekey=11180

[13] U. Shrawankar and V.M. Thakare, "Techniques of feature extraction in speech recognition system: A comparative study", *International Journal of Computer applications in engineering, technology and sciences(IJCAETS)*, ISSN: 0974-3596, pp 412-418, 2010.

[14] Mirlab.org, Audiosignalprocessing [Online]. Available: http://mirlab.org/jang/books/audioSignalProcessing/speechFeatureMfcc.asp?title=12-2%20MFCC

[15] M.A. Anusuya and S.K. Katti, "Speech recognition by Machine: A review", *International journal of computer science and information security(IJCSIS)*, ISSN: 1947-5500, Vol.6,No.3,pp. 181-205, 2009.

[16] G. Nijhawan and M.K. Soni, "Speaker recognition using MFCC and vector quantisation", *International Journal on recent trends in engineering and technology(ACEEE),* Vol.11, No.1, pp. 211-218, 2014.

[17] A. Sizov, K.A. Lee and T. Kinnunen, Unifying Probabilistic Discriminant Analysis variants in biometric authentication. In: Fränti P, Brown G, Loog M, Escolano F, Pelillo M. (eds) Structural, Syntactic, and Statistical Pattern Recognition. S+SSPR 2014. Lecture Notes in Computer Science, vol 8621. Springer, Berlin, Heidelberg. ISBN: 978-3-662-44414-6.

[18] M. Awad and R. Khanna, "Support vector machines for classification", *Efficient learning machines*, Berkeley, CA, Apress, online ISBN: 978-1-4302-5990-9, 2015, pp. 39-66.

[19] SYRIS Technology,2004, Technical document about FAR, FRR and EER. [online]. Available: ftp://syris.com/SYRIS_ACS_DVD-ROM/UserGuideManual/Reader/SYRDF5/About%20FAR_FRR_EER.pdf

[20] MachinelearningCorner[Online]Available: https://mlcorner.wordpress.com/tag/specificity/

[21] LibriSpeech ASR corpus [Online]Available: http://www.openslr.org/12

[22] A. Rozi, D. Wang and Z. Zhang, "An Open/free database and benchmark for Uyghur speaker recognition", *in proceedings thugy20_sre_2015,*)-COCOSDA'15, 2015.

[23] Zeroth - Korean[Online]. Available: http://www.openslr.org/40

[24] F. Rudzicz, A.K. Namasivayam and T. Wolff, (2012) The TORGO database of acoustic and articulatory speech from speakers with dysarthria. *Language Resources and Evaluation*, 46(4), pp. 523-541.

APPENDICES

(Contents of Electronic Disk-CD)

The contents of the CD consist of MATLAB code folders and files and the databases which used for the results obtained and the detailed explanation are as follows:

1. **Real-Time Speaker Identification Folder**: This folder contains the library folder, bleep2 wav file, sound_database (.dat file), speakerrecognition.m main script MATLAB file, T, UBM and V (.mat files) which are used in the implementation of real-time speaker identification.

   i. Library folder: Contains the function.m files which are created and used for the main program.

   ii. Bleep2.wav: is a wav file which is used to play during the multiple process in the real-time identification.

   iii. Sound_database: It is a .dat file, created and used during the process for different variables. In MATLAB, data from the file can be accessible either to write or to read. Data from this file returns as a matrix, multidimensional array or scalar structure array depending on the characteristics of the file. Based on the file format of the input file, *importdata* calls a helper function to read the data. When the helper function returns more than one nonempty output, *importdata* combines the outputs into a *struct* array. This table lists the file formats associated with helper functions that can return more than one output, and the possible fields in the structure array,

File Format	Possible fields	class
MAT-files	One field for each variable	Asoociated with each variable.
ASCII files and spreadsheets	Data, textdata, colheaders, rowheaders	For ASCII files, datacontains a double array. Other fields contain cellarrays of character vectors. textdataincludes row and column headers. For spreadsheets, each field contains a struct, with one field for each worksheet.

iv. Speakerrecognition.m file: This is the main scripting code of MATLAB file, which includes and uses all of the mentioned data.

v. T, UBM and V files: These are empty database with .mat file extensions in the beginning with dimensions 200x832 single, 1x1 struct and 200x65 double respectively and used to store and access the data during execution and it works as per the above table.

Execution: To obtain the results by execution, run the speakerrecognition.m file.

2. **SI_kk_Euclidean:** As per the main aim of the research, Speaker identification comparison between Euclidean Distance and PLDA, this folder consists of one of those, i.e., Euclidean Distance Speaker Identification files with Korean language database. This folder contains the following:

   i. Library

   ii. T, UBM and V (.mat files)

   iii. ED: The database files folder which contains Train, FRR and FAR folders which further consists of .wav sample files.

   iv. Train_Database: The first main MATLAB code file, which runs to train the train signals databases.

   v. Test_FRR and Test_FAR: These are the second main MATLAB code files which runs to train the test signals database.

   vi. Graph_Results.m : This is the MATLAB main code file, which runs to obtain final comparison and results by the data obtained by train and test signals.

   Execution: To obtained the final results, first run Train_Database, then Test_FRR and Test_FAR and then Graph_RESULTS code file.

3. **SI_kk_PLDA:** For the other comparison method PLDA, this folder contains the processing files. Besides, this folder is for Korean language database and the folder contents explains as follows:

   i. Library

   ii. T, UBM and V

iii. Src and two-cov : these folders are also the library folders but which are used for only PLDA implementation. Src folder contains function MATLAB files for the PLDA execution for simp and std variants and the two-cov consists of library functions for two-covariance variants.

iv. Database: consists of the folders of the enrol, train and test signals folders.

v. Databases_enroll.m: The main code to train for the features of enrolled speakers.

vi. Databases_train.m: The main code to train for the features of train samples of the speakers with different utterances.

vii. Databases_test.m: like other two above, to train the test samples of the speakers with multiple utteerances.

viii. Run_PLDA.m: it is the main code to run for the results and plots by the obtained features vectors.

Execution: To get the results, first run Databases_train.m, then Databases_enroll.m, next Databases_test.m and finally Run_PLDA.m main code file.

4. **SI_Libri_Euclidean and SI_UYG_Euclidean:** These two folders also contains the same as SI_kk_Euclidean but just differ in the database. In this, the SI_Libri_Euclidean folder's folder database consists of Libri Speech English language speaker's signals and SI_UYG_Euclidean folder's folder database consists of Uyghur language speaker's signals and the rest information is as same as SI_kk_Euclidean.

5. **SI_Libri_PLDA and SI_UYG_PLDA:** These two folders also contains the same as SI_kk_PLDA but just differ in the database. In this, the SI_Libri_PLDA folder's folder database consists of Libri Speech English language speaker's signals and SI_UYG_PLDA folder's folder database consists of Uyghur language speaker's signals and the rest information is as same as SI_kk_PLDA.

   All functions and few of the main code lines are taken from github.com from pedrocolon93/ivectormatlabmsrit and added so many information according to the requirement of this research and improved accordingly. It is not just the copy of the codes and altered and added so many logical tasks and additional algorithms.

6. **Voice Disorder:** This is regarding the second aim of the study that is to detect the voice disorder and the implementation follows these steps as the folder contains the following:

i. Library
ii. T, UBM and V
iii. TT: It is the databases folder where it consists of the normal (Libri Speech) and the Abnormal speech signals in the as named folders and the test folder contains both samples for the test purpose.
iv. Svm_training.m: This is one of the main code of the experiment where it trains the normal and abnormal samples of the speakers for the training purpose.
v. Svm_test.m: It is also the main code which trains the test samples for the experiment.
vi. SVM_Results.m: This is the final main code which consist of the comparison algorithm and return the output.
vii. Result.m is the functional MATLAB file where it consists of the equations and calculations for the performance measures.

Execution: To get the results out of this implementation, first run the svm_training.m file then svm_test.m file and finally SVM_Results.m code file.

This code is original as of my knowledge because of the use of MATLAB toolbox for the usage of Support vector machine and its classification. One can find complete details about this algorithm in MATLAB's website: https://ch.mathworks.com/