Abstract
Even though speaker recognition is a broad subject, the commercial and personal use implementations are rare. We have to solve several problems before speaker recognition can become more useful. The amount of pattern matching and feature extraction techniques is large and the decision on which ones to use is debatable. One of the main problems of speaker verification in general is the impact of noise. The very popular feature extraction technique MFCC is inherently sensitive to mismatch between training and verification conditions. MFCC is used in many speech recognition applications and is not only useful in text-dependent speaker verification. However the most reliable verification techniques are text-dependent. One of the most popular pattern matching techniques in text-dependent speaker verification is DTW. The signal processing techniques, MFCC and DTW are explained and discussed in detail along with a Matlab program where these techniques have been implemented. The choices made in signal processing, feature extraction and pattern matching are determined by discussions of available studies on these topics. The results indicate that it is possible to program text-dependent speaker verification systems that are functional in clean conditions with tools like Matlab.

Keywords
Speaker Verification, Text Dependent, MFCC, DTW, Feature Extraction, Matlab

I. Introduction
Speaker recognition is classified into two sections namely speaker identification and speaker verification. Verification is the process of automatically identifying who is speaking depending on individual information included in sound waves. This technique helps speaker’s voice sample to verify its identity and control access to various works. Speaker verification is widely applicable while using speaker’s voice sample to verify its identity and control access to services such as Telephone banking, database accessing works, mobile purchasing, voice mailing, security control for secret information fields, and remote access
The human speech contains number of discriminative features that can be used to recognize speakers. Speech contains sufficiently greater frequency up to 5 kHz. The purpose of speaker recognition is to extract features, characterize them and recognize the information about identity of speaker. The property of speech signal changes as a function of time. The time varying Fourier representation is used to analyze the various spectral properties of different speech signal. However, the different properties of speech signal such, as energy, zero crossing, correlation etc. That is its characteristics are stationary for small duration. Hence, with the use of hamming window, Speech sample is divided into a number of frames of small duration so that normal Fourier transform can be used. For this, the Mel frequency Cepstrum Coefficient features are used for designing a text dependent speaker recognition system. In this phase the speaker verification system records the pass-phrase spoken by the user and extracts the MFCC features and records them as the voice-print of that user. Training is used to improve the recognition performance of speaker verification systems by recording several samples of the passphrase instead of just one. When using DTW though there exists a tradeoff between computational efficiency and recognition accuracy. The reason behind this is that creating many templates for a person’s passphrase will increase the number of DTW paths that require to be computed each time before a decision is made [1].

Fig. 1: A Typical Speaker Recognition System

II. Proposed Speaker Recognition System
A speaker recognition system generally consists of five major blocks. These are digital speech input, feature extraction, pattern matching, decision making and enrollment to generate speaker verification systems.

The very first step is to register users that are using the authentication system. This is done by allowing the person to speak a pass-phrase. This analog voice signal is recorded by a microphone and gets converted to a digital signal through the use of an A/D-converter.
The second step is to extract features from the voice signal and saving it to a database. The extracted feature vector is called a voice-print. Each user will have its own voice-print, also called speaker model, in the database that will be used as a reference for the speaker verification task later. Various speaker recognition systems increase their performance by training each user so as to improve the recognition performance of speaker verification systems. The second step is to extract features from the voice signal and saving it to a database. The extracted feature vector is called a voice-print. Each user will have its own voice-print, also called speaker model, in the database that will be used as a reference for the speaker verification task later. Various speaker recognition systems increase their performance by training each user so as to improve the speaker model.
Verification of a each user is carried out by allowing the person utter the pass-phrase belonging to the identity he is claiming into the microphone. The analog voice signal is then converted to a digital signal and when the features get extracted they are compared with the voice-print from the database with the help of a pattern matching. The comparison will give a match score that will be used for checking that the user is accepted or rejected. To
stop tape recorder attacks in text-dependent speaker recognition systems it is necessary to have variety of pass-phrases that can be randomly prompted at verification.

III. MEL Frequency Cepstral Coefficient
There are various ways of extracting features from a voice or speech sample. The most important properties of useful feature extraction methods should be:

• Higher inter-speaker variation
• Lower intra-speaker variation
• Easy for measurement
• Robust against disguise and mimicry
• Robust against distortion and noise
• Maximum internally independent features

One of the most efficient and popular way is to use MFCC. It was first developed for speech recognition but is now one of the most common feature extraction techniques in speaker recognition nowadays. The purpose behind using MFCC is the assumption that the human hearing is an optimal speaker identifier, but it is not yet been confirmed by studies.

The MFCC feature extraction process includes six major steps performed in the following order:
1. Framing
2. Windowing
3. Discrete Fourier Transforming
4. Mel-Frequency Warping
5. Log Compression and Discrete Cosine Transforming
6. Calculation of Delta and Delta-Delta Coefficients

A. Framing
The primary step is framing in which speech signal is split into number of frames typically with the length of 10 to 30 milliseconds. The frame length is essential due to the tradeoff between time and frequency resolution. For frame length to be longer it will be unable to capture local spectral properties and if it is too short there will be degradation of frequency resolution. The frames overlap each other typically by 25% to 70% of their own length. Overlapping ensures that each speech sound is approximately centered at some frame.

B. Windowing
After dividing the signal into frames each frame is multiplied by a specific window function. A required window function should be having a narrower main lobe and a small side lobe. A smooth tapering along the edges is required to reduce discontinuities. In general window used for speaker recognition is the Hamming window.

The Hamming window is defined as:
$$W[n] = 0.54 - 0.46 \cdot \cos \left( \frac{2\pi n}{N} \right) \quad 0 \leq n \leq N - 1$$

The magnitude and frequency response are as below.

C. Discrete Fourier Transforming
The next step is to apply the discrete Fourier transform on each frame.

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{2\pi i k n}{N}}, k = 0, \ldots, N - 1$$

For calculation of DFT faster way is to select FFT which is an algorithm that speeds up DFT Calculations by a hundred-fold. [2]

D. Mel-Frequency Warping
The Mel scale is based on how the human hearing recognizes frequencies. It is defined by putting 1000 Mels equal to 1000 Hz as a reference point and listeners are permitted to adjust the physical pitch until they perceived it as two-fold ten-fold and half fold etc. and that frequencies are labeled as 2000 Mel, 10000 Mel and 500 Mel respectively. The scale is called as Mel scale and is approximately linear below frequencies of 1000 Hz and logarithmic above [14].
The Mel frequency is expressed by the following equation:

$$mel(f) = 2595 \cdot \log_{10}(1 + \frac{f}{700})$$

Where $f$ is the actual frequency and $Mel(f)$ is the perceived one. [3]

The Mel frequency warping while calculating MFCCs is finished by the use of a triangular Mel spaced filter bank. It consists of various triangular shaped and Mel spaced filters whose outputs are described by [3]:

$$Y(i) = \sum_{j=1}^{N} S_j H_{ij}$$

Where $S_j$ is the N-point magnitude spectrum and $H_{ij}$ the sampled magnitude response of an M-channel filter bank and then Mel frequency filter bank is applied in the frequency domain [3].

**E. Log Compression and Discrete Cosine Transforming**

The fifth step is applying compression using logarithm on the filter outputs $Y(i)$ and apply the discrete cosine transform which yields the MFCCs $c[n]$ according to the following formula [4]:

$$c[n] = \sum_{i=1}^{n} \log(Y(i)) \cdot \cos\left(\frac{\pi n}{M}(i-\frac{1}{2})\right)$$

**F. Calculation of Delta and Delta-Delta Coefficients**

Delta and delta-delta coefficients are used to add time evolution details of information and the outcome of first order differentiation is called delta coefficient and the second order derivative is called delta-delta coefficient [3].

The simplest way is using differentiation. The $n$-th delta feature is represented by:

$$\Delta f_k[n] = f_{k+M}[n] - f_{k-M}[n]$$

The $n$-th order delta-delta feature is given by:

$$\Delta^2 f_k[n] = \Delta f_{k+M}[n] - \Delta f_{k-M}[n]$$

where $M$ typically is 2 to 3 frames. The differentiation is done separately on every feature vector [2].

**IV. Enrollment, Training and Verification**

**A. Enrollment and Training**

In this part the speaker verification system records the pass-phrase uttered by the user and extracts the MFCC features and stores them as the speech input of particular user. Training is used to improve the recognition performance of speaker verification systems by recording several utterances of the passphrase. When DTW is used for warping there exist a tradeoff between recognition accuracy and computational efficiency. The main cause behind this is creation of several templates for a person’s pass-phrase will increase the number of DTW paths that need to be computed every time before a decision is made [1].

**B. Pattern Matching**

In the verification process there is necessity of comparing speaker models for determination of similarity, i.e. determining they are from the same speaker or not. The most popular pattern matching technique for text-dependent speaker verification systems is dynamic time warping. It uses template based speaker structure. [5]. Dynamic Time Warping is used to measure similarity of two time series that may vary in time and speed [6].

**C. Decision Process**

The last step is the decision process and is based on the DTW distance computed. The lower the DTW value of distance, the higher will be the score of matching. The decision is taken by:

$$\text{match score}(S_j, VP_j) = \begin{cases} \Theta_j, & \text{accept speaker } j \\ \Theta_j, & \text{reject speaker } j \end{cases}$$

Where $\Theta_j$ is the verification threshold, $S_j$ the speaker voice-print which is compared to the voice print $VP_j$ from the database. The verification threshold can be set same for all speakers or it can be depending on speaker [2].

The threshold is determined by finding a balance between the false rejection rate (FRR) and the false acceptance rate (FAR) of the system, the previous meaning rejecting a valid speaker and the latter accepting an attacker. The balance of these two errors is application dependent [2].
The speaker verification system was implemented in Matlab with the help of several functions provided by Voice box. The most important function was melcepst.m. It calculates the MFCCs of a signal. This function calls on several other functions also included in Voice box. Dynamic Time Warping was implemented by using a source code named dtw.m. It calculates the DTW distance between two vectors. Several built in Matlab functions are also used, which includes various functions for the graphical user interface (GUI) and transforming calculations.

Fig. 8: Graphical User Interface

Add: Adding new user to the list and initiating its variables.
Train: Adding various voice-print templates of the uttered pass-phrase to the user’s database and adjusting the feature vector selection.
Verify: Verifying the user by comparing the uttered pass-phrase to the pass-phrase templates in the user database.
Delete: Removing a user from the list and the database.

There are several ways to reduce the impact of noise on speaker verification systems. These include feature compensation methods, model compensation methods, score compensation methods, filtering techniques, noise compensation, use of microphone arrays and missing feature approaches. Since there are so many methods of improving verification performance in general and its noise robustness there will only be suggestions on how to improve the MFCC feature extraction and the DTW classification procedure.

A. MFCC

One promising approach to improve MFCC performance is to use pitch synchronous feature extraction which uses flexible segmentation to reduce spectral mismatch between training and verification. Various studies shows that it can improve the error rate of a text-dependent speaker verification system using MFCC combined with GMM by 26.7%. The inherent problem of MFCCs, being prone to performance degradation in noisy environments, is partly dependent on the log function used in the MFCC calculation process. To improve this inherent deficiency the log function can be replaced by a combination of a power law function together with the log function. An alternative solution is to replace the log function with a root function which has also shown to increase noise robustness. Due to the uneven sensitivity of MFCC features to mismatches in the environment it is also possible to use weighting methods, which are attractive because they only need a trivial change in the decoding methodology.

B. DTW

This method, combined with MFCC feature extraction, improves single digit recognition rate from 85.3% to 99% when comparing the use of a single template and a 40-template CWRT, according to a study. An interesting and simple improvement to DTW is time scale modification (TSM). Studies show that TSM reduces the computational complexity of DTW and at the same time increases the noise robustness of recognition systems. TSM is achieved by shrinking test and reference pass-phrase on the time axis by a certain optimal factor. Studies show that MFCC feature extraction combined with DTW and TSM can reduce the computational complexity up to 75% in clean conditions without affecting the recognition performance.

C. Other improvements

There are possible issues with the sample-rate in the A/Dconversion. Based on the facts discussed in that chapter it is suggested that a sampling frequency of 11025Hz should be used instead of 8 KHz. Another improvement would be to only record a voice sample when a person is speaking into the microphone instead of always recording a 5 second long sample after the train or verification button is pressed. This would decrease the risk of contaminating the voice sample with accidental noise in training and verification mode.

VII. Conclusion and Future work

The result of the reviewed studies on speaker verification yielded the answer that MFCC and DTW work well together for text-dependent speaker verification purposes. Together with smaller adjustments and improvements of the weak spots of these two techniques, it can be concluded that a fully operational speaker verification program can be developed in a Matlab environment. Even though feature extraction and pattern matching are core functions of a speaker verification system there was a surprising amount of lateral problems to understand. Where this thesis project ends another one can begin. There are several ways of expanding this system for example by including speaker identification functionality and adding solutions for noise robustness. A close relative to DTW is dynamic time warping (DDTW). An inherent weakness of DTW is that it only considers Y-axis values when determining the alignment of two signals. This gives unintuitive alignments where a single point of one of the signal is mapped onto a large section of the other. To avoid such behavior DDTW is used instead. Instead of considering Y-axis values for alignment it considers the shape by using the local derivatives. This way DDTW can achieve superior alignments compared to the ordinary DTW technique.

An interesting replacement of the widely used feature extraction technique MFCC is the wavelet transform. Unlike MFCCs it doesn’t lack the ability to resolve the temporal characteristics of the signal. Another interesting alternative is the Zak Transform which is an alternative with a model and identification complexity advantage.
VIII. Acknowledgement

To complete this work, we have got valuable suggestions and guidance from different experts of this field. All the people I had interesting discussions with about my work and everyone that helped me test my system. I am also thankful to all the researcher and authors of the manuscript from where I got valuable information for completing the work.

References


Vijaykumar V. Patil received his B.E. degree in Electronics Engineering from D Y Patil College of Engineering, Kolhapur affiliated to Shivaji University, Kolhapur in 1995 and M.E. degree in Electronics Engineering from Tatyasaheb Kore Institute of Engineering and Technology, Warananagar affiliated to Shivaji University, Kolhapur in 2009. He is currently working as an Assistant Professor in KIT College of Engineering, Kolhapur affiliated to Shivaji University, Kolhapur from 2003.

Kaustubh B. Joshi received his B.E. degree in Electronics and Telecommunication Engineering from Finolex Academy of Management and Technology, Ratnagiri affiliated to University of Mumbai, in 2013 and Pursuing M.E. degree in Electronics and Telecommunication Engineering from KIT College of Engineering, Kolhapur affiliated to Shivaji University, Kolhapur from 2013. He is currently working as a Teaching Assistant in Maharshi Parshuram College of Engineering, Velneshwar affiliated to University of Mumbai from 2014.