Vol II Issue XII June 2013

Impact Factor : 1.2018

ISSN No :2231-5063

Monthly Multidiciplinary Research Journal

Golden Research Thoughts

Chief Editor
Dr.Tukaram Narayan Shinde

Publisher
Mrs.Laxmi Ashok Yakkaldevi

Associate Editor
Dr.Rajani Dalvi

Honorary
Mr.Ashok Yakkaldevi

IMPACT FACTOR : 1.2018

Welcome to ISRJ

RNI MAHMUL/2011/38595 **ISSN No.2230-7850**

Indian Streams Research Journal is a multidisciplinary research journal, published monthly in English, Hindi & Marathi Language. All research papers submitted to the journal will be double - blind peer reviewed referred by members of the editorial Board readers will include investigator in universities, research institutes government and industry with research interest in the general subjects.

Address:-Ashok Yakkaldevi 258/34, Raviwar Peth, Solapur - 413 005 Maharashtra, India
Cell : 9595 359 435, Ph No: 02172372010 Email: ayisrj@yahoo.in Website: www.isrj.net

Golden Research Thoughts
Volume 2, Issue. 12, June. 2013
ISSN:-2231-5063
Available online at www.aygrt.isrj.net

ORIGINAL ARTICLE

SPEAKER RECOGNITION USING MATLAB

PRIYANKA ASHOK SHISODE AND VIDYA SHANKAR MISKIN

[1,2]Department Of Computer Sciences Solapur (Maharashtra), India.

Abstract:

Speaker recognition is the project build in matlab. Speaker recognition is used to recognize the speaker's identity. As we know every human being has a unique voice so, just by hearing, it is possible to recognize the particular person. Speaker recognition uses the natural features of speech that patterns reflect both size and shape of mouth & throat and speaking style. Speaker recognition uses algorithms for extracting features from speech signals. That features can be pitch, amplitude, frequency etc. The main purpose of this project is to match the sample of training phase with the testing phase

KEYWORDS:

Speaker Recognition, Feature Extraction, Mel Frequency Cepstral Coefficients, Vector Quantization.

INTRODUCTION :

This project is developed in matlab. Methods of speaker recognition are a text dependent and text independent. Text dependent depends on predermined statements. A text dependent system relies on the restriction on the text. In the text dependent recognition Hidden Markov Model (HMM) are used. Dynamic Time Wrapping (DTW) stores the training vector sequences without any further processing. The text dependent is a combination of DTW and GMM approach. Example of text dependent is a user's password for authentication system. Text independent means can identify the speaker of what is being said. In a text independent speaker recognition system does not have any information about the content. For text independent recognition, Gaussion Mixture Model (GMM) or Vector Quantization is used. The benefit of using GMM is a well understood statistical model. Speaker recognition can divide into two parts like speaker identification and speaker verification. In the Speaker identification is a task of determine the unknown speaker identity. In the speaker verification is a task of accepting or rejecting speaker. Speaker recognition system contains two module feature extraction and feature matching. In the features extraction is a process of extract the feature of speech signal using mfcc.Feature matching involves the actual identify the unknown speaker by comparing extracted feature from set of known speaker.

There are some applications where speaker recognition plays important role that are authentication, surveillance and forensic speaker recognition. In the speaker recognition authentication, allows the user to identify themselves. Speaker recognition for surveillance used in security agencies has several means of collecting information. One of these is electronic eavesdropping of telephone and radio conversations. In the forensic speaker recognition, used for recorded voice can help to convict a criminal or discharge on innocent in court. To identify the voice of the unknown speaker we need following things:

- Extract feature of the speech of the known speakers.
- Create a feature model of the known speaker.
- Matching the features from both the known and unknown speakers using statistical models of the

voice.

- Make decision when we identify the speakers.

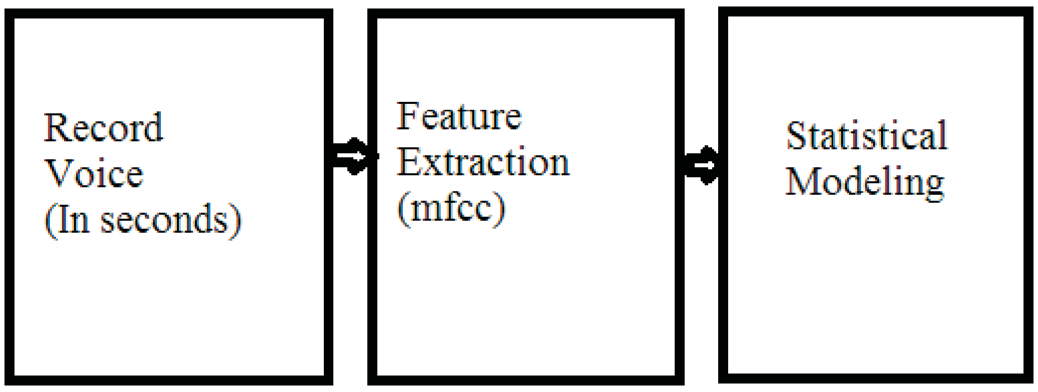

Figure (a): Block diagram of recording voice.

In this project we have used statistical model , feature extracted from speech signals for identify the speaker. We are giving the facility to record the voice in seconds and then identify the speaker. We will discuss each module in detail. The following figure (a) block diagram show the flow of recording voice. First, record voice by alloting time (in seconds), by using mfcc technique extract the feature of recorded voice and display statistical model with the help of frequency. The recorded voice can be saved wherever you want. Figure (b) show the example of recorded voice.

We analyzed the sample of 'test\s1.wav'.

```
[s1, fs1] = wavread('test\s1.wav');
```

This command passes the signal data into vector s1 and variable fs1 stored the sampling frequency which is 8000 by default. (Or 8000 sample per seconds) According to the Shannon-whithake theorem: [4]

Ws = 2* Wo => fs = 2 * fo =>fo = fs / 2 = 8000 / 2 = 4000[Hz]

One sample represents 1 / fs[s]. 256 samples therefore contain 256 / fs[s] of the actual signal.

256 / 8000 = 0.032[s] = 32.00[ms].Plot the signal to view it in the time domain. So the goal of this step should be clear.

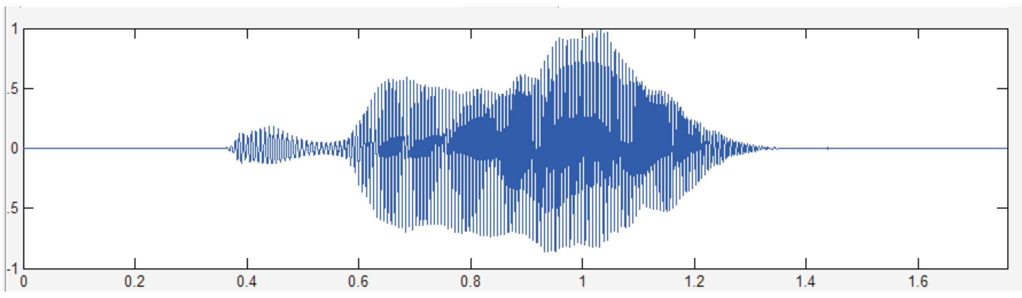

Figure (b): Statistical model of waveform.

Another facility given is to identify the unknown speaker. In the speaker recognition we used two phases i.e training phase and testing phase. Figure (c) here is just overview of the project, first extract feature form the training phase and extract feature from the testing phase using mfcc technique. After extracting features from both the phases it is possible to find the matches. We can developed these project using matlab tools i.e. Feature extraction using MFCC (Mel Frequency Cepstral Coefficients) and Statistical modeling using Vector Quantization.

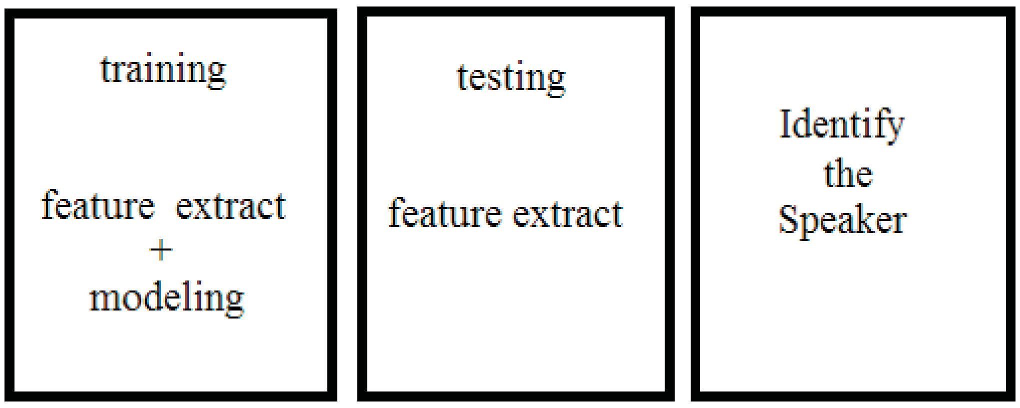

Figure (c): block diagram of speaker recognition.

We recognize the speaker from already existing wave files in the train and test folder. There are eight wave files in the train folder name as s1, s2... s8. All speakers speak same single digit "Zero" similarly in test folder. Play each sound file in the train folder and test folder. You can easily identify the speaker because every speaker has unique identity.

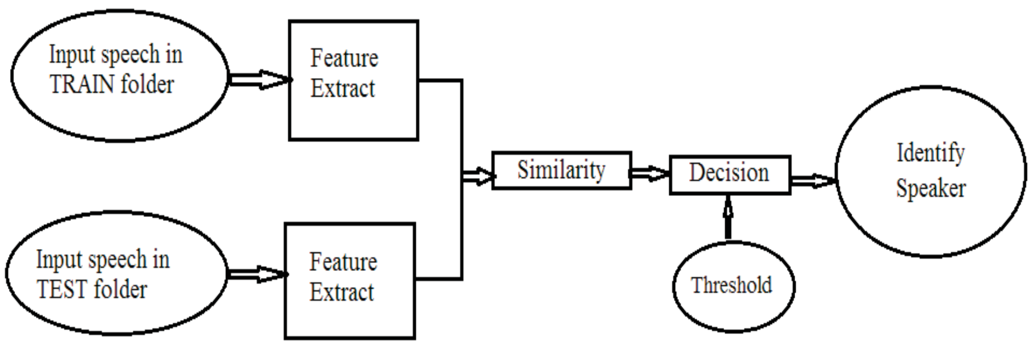

Figure (d): Structure of the speaker recognition.

Figure (d) show the task to be performed in step by step.Frist, read speech from the train and test files then feature extraction from both the train and test files and create statistical model of the train files then matches the each test files with the models created. After you can identify the matches of speakers

2. Methodology

Speaker Recognition uses different algorithms to match the voice of a speaker. It uses algorithms like Linear Prediction Coding (LPC), Mel-Frequency Cepstrum Coefficients (MFCC), and others. Linear prediction is a method for signal source modelling dominant in speech signal processing and having wide application in other areas. But we used the Mel-Frequency Cepstrum Coefficients (MFCC). These is well-known in the field of speech recognition also, therefore, they can be regarded as the "standard" features in speech as well as speaker recognition.

2.1 Feature Extraction

Using digital signal processing the voice is converted to speech waveform. This is called signal processing. As the sound changes it also changes the speech signals. The following figure (e) show the example of speech signal.

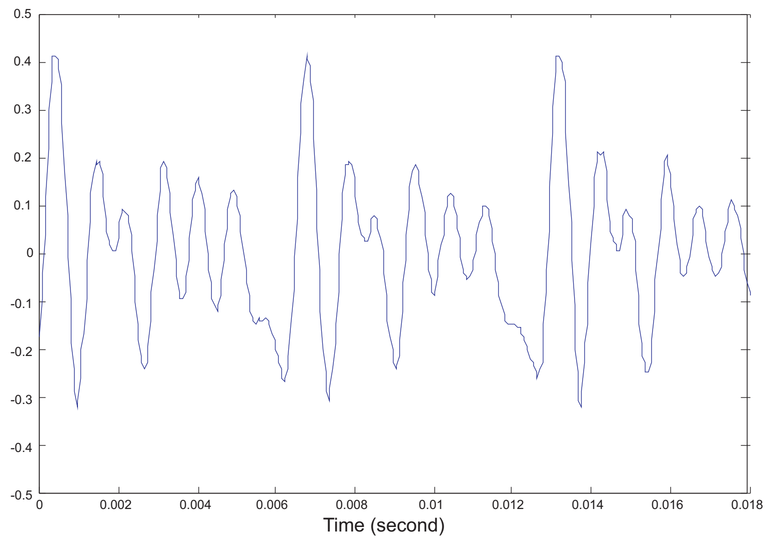

Figure (e): Example of speech signal.

2.2 Mel Frequency Cepstral Coefficients (MFCC's):

MFCC uses Fourier transform (FFT).This are coefficients that represent audio. The main purpose of the MFCC processor is to act the behavior of the human ears. MFCC process is divided into five blocks. These are as follows.

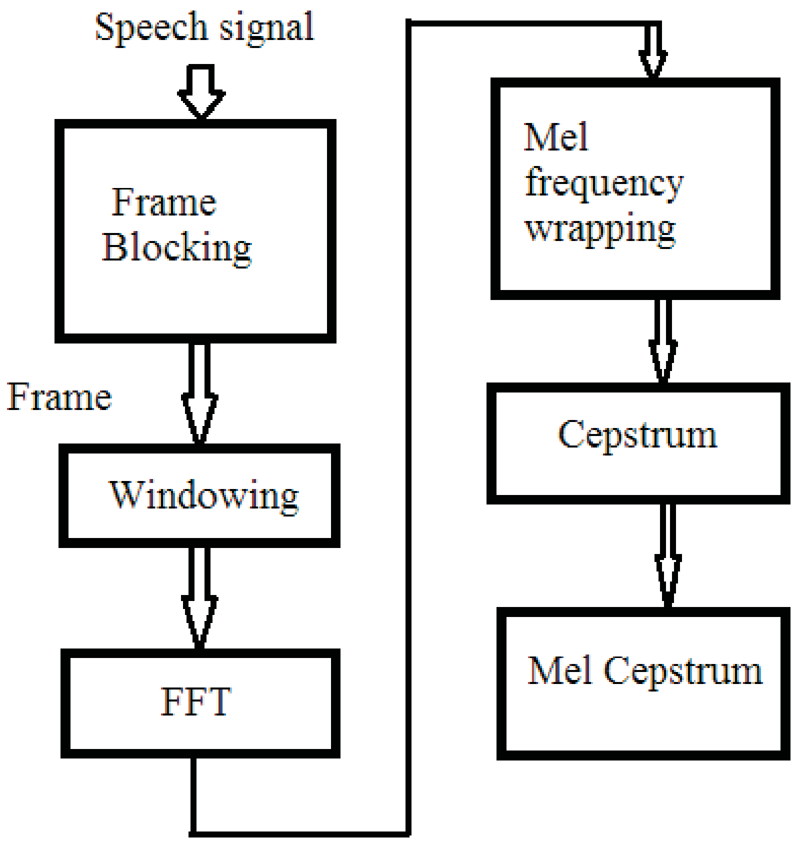

Figure (f): Block diagram of MFCC

2.2.1 Frame Blocking:

In this step the continuous speech signal is blocked into frames of N samples, with adjacent frames being separated by M (M < N) [3]. The first frame consists of the first N samples. The second frame begins M samples after the first frame, and overlaps it by N - M samples and so on [3]. This process continues until all the speech is accounted for within one or more frames. Typical values for N and M are N = 256 (which is equivalent to ~ 30 msec windowing and facilitate the fast radix-2 FFT) and M = 100 [3].

The feature recognition process cuts the digitized audio signal, i.e. the sequence of sample values, into overlapping windows of equal length.The cut-out portions of the signal are called "frames", they are extracted out of the original signal every 10 or 20 ms. The length of a frame is about 30 ms. For speaker recognition tasks, sometimes longer frames are used in comparison to the feature extraction method used for speech recognition in order to increase spectral resolution. Each frame in the time domain is transformed to a MFCC vector.

In the following code, s1 is the signal to analyze, l is used to store the length of signal, and m is the distance between the beginnings of two frames, n is the number of samples per frame.

To obtain a matrix M containing all the frames, we used the following script: [4]

```
l = length(s1);

m = 100;

n = 256;

nbFrame  =  floor((l - n) / m) + 1;

for i = 1: n
```

2.2.2 Windowing:

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. If we define the window as

$$y_l(n) = x_l(n)w(n), \quad 0 \le n \le N-1$$

Typically the Hamming window is used, which has the form [3]:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \le n \le N-1$$

```
h = hamming (n);

M2 = diag (h) * M;
```

This reduces the size of the signal at the edges. We need a shape which has a spectrum with a narrow central lobe and small sidelobes. We use n i.e number of samples per frame is to create a hamming matrix which is used to create a new matrix M2.

2.2.3 Fast Fourier Transform :

The next processing step is the Fast Fourier Transform, which converts each frame of N samples from the time domain into the frequency domain. The FFT is a fast algorithm to implement the Discrete Fourier Transform (DFT), which is defined on the set of N samples $\{x_n\}$, as follow [3]:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi kn/N}, \qquad k = 0,1,2,\ldots,N-1$$

In general X_k's are complex numbers and we only consider their absolute values (frequency magnitudes). The resulting sequence $\{X_k\}$ is interpreted as follow: positive frequencies $0 \le f < F_s/2$ correspond to values $0 \le n \le N/2-1$ while negative frequencies $-F_s/2 < f < 0$ correspond to $N/2+1 \le n \le N-1$ Here, Fs denotes the sampling frequency[3].

The result after this step is often referred to as spectrum or periodogram.

```
for i = 1:nbFrame

   M3(:, i) = fft(M2(:, i));
```

[4] We create a new matrix M3 where the column vectors are the FFTs of the column vectors of M2.

2.2.4 Mel-frequency Wrapping:

In this each tone with actual frequency is measured on the scale called "mel" scale. The mel-frequency scale is linear frequency spacing below 1000 Hz and a logarithmic

spacing above 1000 Hz. The main use of mel-wrapping filter bank is to view each filter as a histogram bin in the frequency domain. Where bins have overlap. The number of mel spectrum coefficients, K, is typically choSsen as 16.

2.2.5 Mel-Cepstrum:

In this step we convert the mel-spectrum back to time domain. That is called mel-cepstrum coefficient (MFCC). The cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis. Because the mel spectrum coefficients (and so their logarithm) are real numbers, we can convert them to the time domain using the Discrete Cosine Transform (DCT). Therefore if we denote those mel power spectrum coefficients that are the result of the last step are [3] $\tilde{S}_0, k = 0,2,\ldots,K-1$, we can calculate the MFCC's $\tilde{c}_n$, as

$$\tilde{c}_n = \sum_{k=1}^{K} (\log \tilde{S}_k) \cos\left[n\left(k-\frac{1}{2}\right)\frac{\pi}{K}\right], \qquad n = 0,1,\ldots,K\text{-}1$$

Note that we exclude the first component, $\tilde{c}_0$, from the DCT since it represents the mean value of the input signal, which carried little speaker specific information [3].

2.3 Vector Quantization:

This is a text independent technique. Vector quantization is a technique which is also used for speech coding. The training material is used to estimate a code book. This includes mean vectors of feature vector clusters which are given indexes in order to identify them. For compression of speech, the index number of the nearest cluster is used instead of the original feature vector.

This observation is also true in regard to speaker specific codebooks which are used for speaker recognition. The training material of a speaker is used to estimate a codebook, which is the model for that speaker. The classification of unknown test signals [4].

3. Result:

The following result which is occurred by comparing 8 files in the training phase with testing phase. It displays the matches from both the phases.

```
\speaker_recognize\train\s1.wav
\speaker_recognize\train\s2.wav
\speaker_recognize\train\s3.wav
\speaker_recognize\train\s4.wav
\speaker_recognize\train\s5.wav
\speaker_recognize\train\s6.wav
\speaker_recognize\train\s7.wav
\speaker_recognize\train\s8.wav
Speaker 1 matches with speaker 6
Speaker 2 matches with speaker 1
Speaker 3 matches with speaker 4
Speaker 4 matches with speaker 7
Speaker 5 matches with speaker 5
Speaker 6 matches with speaker 3
Speaker 7 matches with speaker 4
Speaker 8 matches with speaker 8
```

4. CONCLUSION:

The purpose of this project identifies the speaker. We transform speech signals into vector. By applying vector quantization (VQ) –based pattern recognition technique. We compared the speakers of training phase with the speaker of testing phase, by comparing its vectors. This is helpful for matching the speaker. MFCC is used to extracting features.This project has facility of recording voice depending on time duration and also display the statistical model based on its frequency. It also reduce external noise.

5. REFERENCES

[1] Anil Alexander, Andrzej Drygajlo"03-Biometrics-Exercise-3-2005" Swiss Federal Institute of Technology, Lausanne. Signal Processing Institute.

[2] E. Darren Ellis"Project"Department of Computer and Electrical Engineering – University of Tennessee, Knoxville, Tennessee 37996.

[3] DSP Mini-Project: An Automatic Speaker Recognition System. http://www.ifp.uiuc.edu/~minhdo/teaching/speaker_recognition

[4] Christian Cornaz, Urs Hunkeler"minipro2" February 3, 2003

[5] G. Suvarna Kumar et. al. / International Journal of Engineering Science and Technology, Vol. 2(6), 2010, 2428-2436.