World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 2, No. 6, 203-208, 2012

Text-Independent Speaker Identification Using Hidden Markov Model

Sayed Jaafer Abdallah
College of Computer Science and Information Technology
Sudan University of Science and Technology
Khartoum, Sudan

Izzeldin Mohamed Osman
College of Computer Science and Information Technology
Sudan University of Science and Technology
Khartoum, Sudan

Mohamed Elhafiz Mustafa
College of Computer Science and Information Technology
Sudan University of Science and Technology
Khartoum, Sudan

Abstract—This paper presents a text-independent speaker identification system based on Mel-Frequency Cepstrum Coefficient (MFCC) feature vectors and Hidden Markov Model (HMM) classifier. The implementation of the HMM is divided into two steps: feature extraction and recognition. In the feature extraction step, the paper reviews MFCCs by which the spectral features of speech signal can be estimated and shows how these features can be computed. In the recognition step, the theory and implementation of HMM are reviewed and followed by an explanation of how HMM can be trained to generate the model parameters using Forward-Backward algorithm and tested using forward algorithm. The HMM is evaluated using data of 40 speakers extracted from Switchboard corpus. Experimental results show an identification rate of about 84%.

Keywords- Speaker identification; MFCC; HMM; Feature extraction; Forward-Backward; and Switchboard.

I. Introduction

Speaker recognition is the process of automatically recognizing who is speaking on the basis of information obtained from speech waves. This technique will make it possible to verify the identity of persons accessing systems, that is, access control by voice, in various services. These services include voice dialing, banking transaction over telephone network, telephone shopping, database access services, information and reservation system, voice mail, security control for confidential information areas, and remote access to computers [1]. Speaker recognition is probably the only biometric which may be easily tested remotely through the telephone network, this makes it quite valuable in many real applications, and it will become more popular in the future [2].

Speaker recognition is divided into speaker verification and speaker identification. For speaker verification an identity is claimed by the user, and the decision required of the verification system is strictly binary; i.e., to accept or reject the claimed identity [3]. Speaker identification is the process of determining which speaker in a group of known speakers most closely matches the unknown speaker [4].

The data used in the recognition is divided into text-dependent and text-independent. In text-dependent, the speaker is required to provide utterances having the same text for both training and recognition [1], whereas the text-independent systems allow the user to utter any text [4].

The steps for identifying the unknown speaker are shown in Fig.1. The observation sequence $O=\{o_1 o_2 \dots o_T\}$ is measured, via a feature extraction and vector quantization; followed by calculation of likelihoods ($P(O|\lambda_s), 1 \le s \le 40$) for all models, then we select the HMM model whose likelihood is highest, i.e., $\max_{1\le s\le 40}\left[P(O|\lambda_s)\right]$.The likelihood probability is computed by using the forward algorithm.

Where *T* is the length of the observation sequence, *s*, is speaker index, and λ_s , is speaker model.

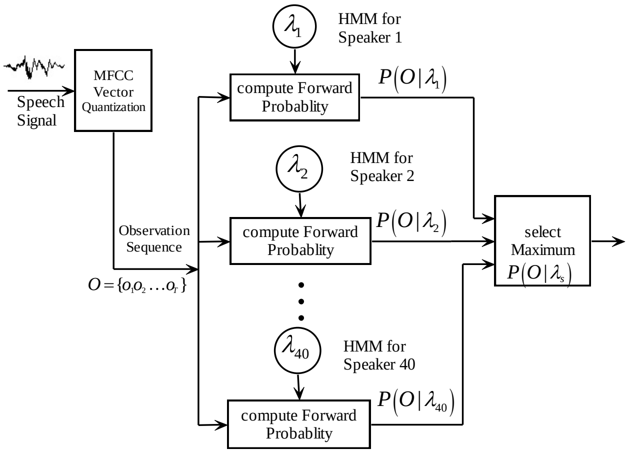

Figure 1. Block diagram of HMM-Based Recognizer (after Rabiner[5]).

II. Mel-Frequency Cepstrum Coefficients

A. *Mel-Frequency*

Psychophysical studies have shown that human perception of the frequency content of sounds does not follow a liner scale. That research has led to the concept of the subjective frequency, i.e., the perceived frequency of sounds is defined as follows. For each sound with an actual frequency, f , measured in Hz, a subjective frequency is measured on a scale called the "Mel scale" [6]. Mel-frequency can be approximated by

$$\mathrm{Mel}(\mathrm{f}) = 1127\ln\left(\frac{f}{700}+1\right), \tag{1}$$

where f in Hz, is the actual frequency of the sound [7].

B. *Cepstrum*

Cepstrum is defined as the inverse Fourier transform of the logarithm of the magnitude of the Fourier transform [8]; i.e.

$$cepstrum = \mathrm{ifft}\left(\log\left(\left|\mathrm{fft}\left(signal\right)\right|\right)\right), \tag{2}$$

where the function iff( ) returns the inverse discrete Fourier transform, and the function fft( ) returns the discrete Fourier transform of signal.

The first application of cepstrum to speech processing was proposed by Noll, who applied the cepstrum to determine the pitch period [8]. The cepstrum used also to distinguish underground nuclear explosions from earthquakes [9].

C. *Triangular Filters Bank*

The human ear acts essentially like a bank of overlapping band-pass filters [9] and human perception is based on Mel scale. Thus, the approach to simulating the human perception is to build a filter bank with bandwidth given by the Mel scale and pass the magnitudes of the spectra, through these filters and obtain the Mel-frequency spectrum [2].

We define a triangular filter-bank with *M* filters (*m*=1, 2,…,*M*) and *N* points Discrete Fourier Transform (DFT) (*k*=1,2,…,*N*), where, $H_m[k]$, is the magnitude (frequency response) of the filter given by:

$$H_m[k] = \begin{cases} 0, & k < f[m-1] \\ \dfrac{\left(k - f[m-1]\right)}{\left(f[m] - f[m-1]\right)}, & f[m-1] \le k \le f[m] \\ \dfrac{\left(f[m+1] - k\right)}{\left(f[m+1] - f[m]\right)}, & f[m] \le k \le f[m+1] \\ 0, & k > f[m+1] \end{cases} \tag{3}$$

Such filters compute the average spectrum around each center frequency with increasing bandwidths, and they are displayed in Fig.2 [7, 10].

Let f_l and f_h be the lowest and highest frequencies of the filter-bank in Hz, F_s the sampling frequency in Hz, *M* the number of filters, and *N* the size of the Fast Fourier Transform. The boundary points, shown in (4) are uniformly spaced in Mel-scale [7, 11].

$$f[m] = \left(\frac{N}{F_s}\right)\mathrm{Mel}^{-1}\left(\mathrm{Mel}(\mathrm{f}_l) + \mathrm{m}\frac{\mathrm{Mel}(\mathrm{f_h}) - \mathrm{Mel}(\mathrm{f}_l)}{\mathrm{M}}\right), \tag{4}$$

$$0 \le m \le M+1$$

where $\mathrm{Mel}(f)$ is given by (1) and $Mel^{-1}(f)$ is its inverse given by (5)[11].

$$\mathrm{Mel}^{-1}(\mathrm{f}) = 700\left(\exp\left(\frac{f}{1127}\right) - 1\right), \tag{5}$$

where f in Mel, is the subjective frequency (Mel-frequency).

We assume that the sampling frequency is $F_s = 8$ kHz , the size of Fast Fourier Transform is *N*=512, and the number of filters *M*=20.

Let $f_l = F_s/N = 8000/512 = 15.6$ Hz , and $f_h = F_s/2 = 4$ kHz be the lowest and highest frequencies of the filter. Using (4), the boundary points of the filter-bank Fig. 2 can be shown as:

$$f[m] = \left(\frac{512}{8000}\right)\mathrm{Mel}^{-1}\left(\begin{matrix}\mathrm{Mel}(15.6) + \\ m\dfrac{\mathrm{Mel}(4000) - \mathrm{Mel}(15.6)}{20}\end{matrix}\right). \tag{6}$$

$$0 \le m \le 21$$

The distance between two critical frequencies is approximately 106 Mels, as in (7) and the width of the triangle is 212 Mels.

$$\frac{\mathrm{Mel}(4000) - \mathrm{Mel}(15.6)}{20} = \frac{2146\text{-}25}{20} = 106 \tag{7}$$

D. *Calculation of MFCCs*

Given the DFT of the input signal, $x[n]$,

$$X_a[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi nk/N}, \quad 0 \le k \le N \tag{8}$$

In most implementations of speech recognition, a short-time Fourier analysis is done first, resulting in a DFT, $X_a[k]$ for the a^{th} frame. Then the values of DFT are weighted by triangular filters [7]. The result is called Mel-frequency power spectrum which is defined as

$$S[m] = \sum_{k=1}^{N} X_a[k]^2 H_m[k], \quad 0 < m \le M \tag{9}$$

where $X_a[k]^2$ is called power spectrum. Finally, a discrete cosine transform (DCT) of the logarithm of $S[m]$ is computed to form the MFCCs as

$$mfcc[i] = \sum_{\mathrm{m}=1}^{\mathrm{M}} \log\left[S[m]\right]\cos\left[\mathrm{i}\left(\mathrm{m}\text{-}\frac{1}{2}\right)\frac{\pi}{M}\right], \tag{10}$$

$$i = 1, 2, \ldots, L$$

where *L* is the number of cepstrum coefficients[8].

The DCT is related to the DFT, and in fact, may be written as function of the DFT. One of the main advantages of the DCT in speech processing is that the transform coefficients are not correlated (are not all of equal perceptual importance) [9].

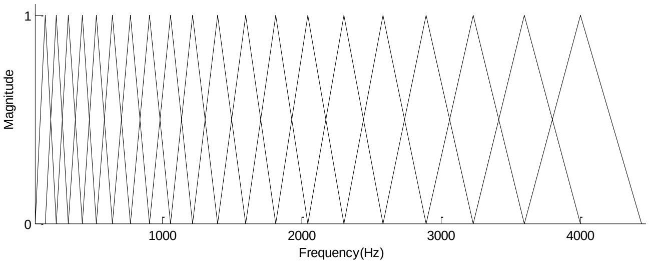

Figure 2. Filter bank for generating Mel-Frequency Cepstrum Coefficients(after Davis and Mermelstein[12]).

III. HIDDEN MARKOV MODELS

A. *Definition of Hidden Markov Model*

Hidden Markov model (HMM) describes a two-stage stochastic process. The first stage consists of a Markov chain. In the second stage then for every point in time t an output or emission (observation symbol) is generated. This sequence of emissions is the only thing that can be observed of the behavior of the model. In contrast, the state sequence taken on during the generation of the data cannot be observed [13].

B. *Elements of an HMM*

An HMM for discrete symbol observations is characterized by the following:

- N, the number of hidden states in the model. We label the states as $N = \{1, 2, ..., N\}$, and denote the state at time *t* as q_t [5].
- M, the number of distinct observation symbols per state. We denote the symbols as $V = \{v_1, v_2, ..., v_M\}$ [14].
- State transition probability distribution, $A = \{a_{ij}\}$,where

$$a_{ij} = P[q_{t+1} = j \mid q_t = i], \quad 1 \le i, j \le N \tag{11}$$

- The observation symbol probability distribution in state $j, B = \{b_i(k)\}$, where

$$b_i(k) = P[o_t = v_k \mid q_t = i], \quad 1 \le k \le M \tag{12}$$

- The initial state distribution, $\pi = \{\pi_i\}$, where

$$\pi_i = P[q_t = i], \qquad 1 \le i \le N \tag{13}$$

For convenience, we use the compact notation, $\lambda = (A, B, \pi)$ to indicate the complete parameter set of the model [6].

C. *Training the HMMs*

For each speaker *s* in the database, we must build an HMM λ_S , i.e. we estimate the model parameters $\lambda = (A, B, \pi)$ that maximize the likelihood of the training dataset. The steps for estimating the model parameters $\lambda = (A, B, \pi)$ are illustrated in Fig. 3.

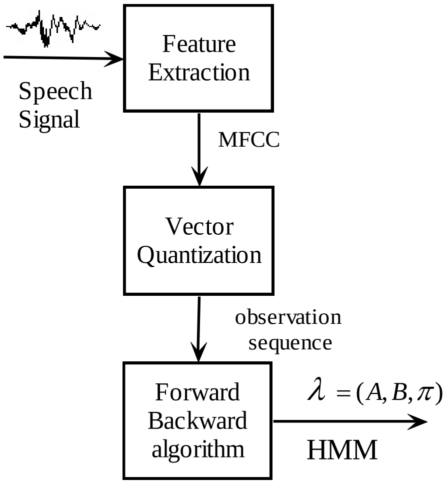

Figure 3. Steps used to estimate the parameters of HMMs.

The input speech signal is converted into vectors of MFCC. Then the feature vectors are quantized into observation sequences. The quantization is achieved by k-mean algorithm and classification procedure.

Vector quantization is required to map each continuous observation vector (MFCC) into a discrete codebook index (or symbols). The resulting symbols are new features which were used as input to estimate the HMM parameters.

Finally, the models parameters are estimated from the observation sequences using the Forward-Backward Algorithm.

D. *Forward and Backward Algorithm*

Consider the forward variable $\alpha_t(i)$ defined as

$$\alpha_t(i) = P\left(o_1 o_2 \ldots, o_t, q_t = i \mid \lambda\right) \tag{14}$$

That is, the probability of the partial observation sequence, $o_1 o_2 \ldots, o_t$ (from 1 until time *t*) and state *i* at time *t*, given the model λ [9]. We can solve for $\alpha_t(i)$ using the following forward algorithm:

1. Initialization

$$\alpha_1(i) = \pi_i b_i(o_1), \quad 1 \le i \le N \tag{15}$$

2. Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}), \tag{16}$$

$$1 \le t \le T-1, \quad 1 \le j \le N$$

3. Termination

$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i), \tag{17}$$

Figure 4. Forward Algorithm (After Rabiner and Juang[6]).

where π_i is an initial transition probability, a_{ij} is a transition probability from state i to state j, and $b_j(o_{t+1})$ is probability of observing the symbol o_{t+1} from state j.

Consider the backward variable $\beta_t(i)$ defined as

$$\beta_t(i)=P(o_{t+1}o_{t+2}\dots,o_T,|q_t=i,\lambda), \tag{18}$$

that is, the probability of the partial observation sequence, $o_{t+1}o_{t+2}\dots,o_T$ (from t+1 until time T) given state i at time t and the model λ [9]. We can solve for $\beta_t(i)$ using the backward algorithm shown in Fig. 5.

1. Initialization

$$\beta_T(i)=1, \qquad 1\le i\le N \tag{19}$$

2. Induction

$$\beta_t(i)=\left[\sum_{j=1}^{N}a_{ij}b_j(o_{t+1})\beta_{t+1}(j)\right], \tag{20}$$

$$t=T-1,\ T-2,\dots,1,\quad 1\le i\le N$$

Figure 5. Backward Algorithm (After Rabiner and Juang [6]).

E. *Estimating the Parameters* $\lambda=(A,B,\pi)$

The model parameters can be estimated as follows [5]:

$$\pi_i=\gamma_1(i) \tag{21}$$

= expected number of times in state i

$$a_{ij}=\frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} \tag{22}$$

$$=\frac{\text{expected number of transitions from state } i \text{ to state } j}{\text{expected number of transitions from state } i}$$

$$b_i(k)=\frac{\sum_{\substack{t=1\\ o_t=v_k}}^{T}\gamma_t(i)}{\sum_{t=1}^{T}\gamma_t(i)} \tag{23}$$

$$=\frac{\text{expected number of times in state } i \text{ and observing } v_k}{\text{expected number of times in state } i}$$

Using the forward $\alpha_t(i)$ and backward $\beta_t(i)$ variables [5], $\xi_t(i,j)$, $\gamma_t(i)$ are defined as

$$\xi_t(i,j)=\frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{P(O|\lambda)} \tag{24}$$

$$\gamma_t(i)=\frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)} \tag{25}$$

IV. Experimental Results

A. *Speech Database*

We used the Switchboard [15] Telephone Speech Corpus which was designed and recorded for text-independent speaker identification and speech recognition. For our experiments, we selected a subset from the Switchboard Corpus. This subset contains 40 speakers (22 males+18 females), with 20 utterances per speaker. About 70% of the utterances were selected randomly to form the training dataset, and the remaining utterances were used as the testing dataset, see table I. The duration of each utterance was 3.2 seconds, see Fig. 6, i.e., the size=50 KB (*The speech was recorded using a sampling rate of 8000 Hz with 16 bits per sample*).

Fig. 6 shows the waveform corresponding to the utterance, "Computer here at the house and I made a Lotus spreadsheet" as spoken by a female speaker.

B. *Feature Extraction*

We used a population of 40 speakers, with 20 utterances for each. Each utterance was divided into 198 frames. The average length of each frame was about 32 milliseconds (256 samples). Then MFCC were calculated for each frame. After performing feature extraction for each speaker, it was determined that each speaker had at least 3960 MFCC feature vectors.

C. *Vector Quantization*

For quantization purpose, the training vectors were used to generate a codebook of length 128 (A codebook of length less than 128 produced degraded results). The codebook was generated by K-mean clustering algorithm. Finally, both training and testing vectors were quantized to generate the observation sequences to be input into the HMMs.

TABLE I. Size of the Dataset Used in the Experiments.

Number of Speakers	***Numbers of utterance per speaker***		***Total***
40	Training	14	560
	Testing	6	240
	Total	20	800

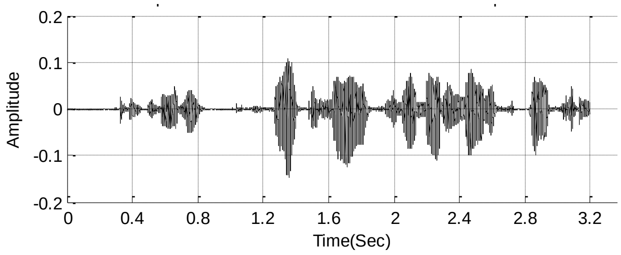

Figure 6. Speech Waveform sampled at 8 kHz.

D. Structure of HMMs

For training experiments, all the training data was used to train 40 hidden Markov based speaker models. All of the 40 models had the same topology 8-states, left-to-right models as shown in Fig. 7. Each model λ , had a transition probability matrix $A = \{a_{ij}\}$, an initial state probability matrix $\pi = \{\pi_i\}$, and an observation probability matrix $B = \{b_i(k)\}$.

The issue of the number of states to use in each model leads to many ideas. Rabiner and Juang[6] proposed 5 to 10 states for each model. The training sequences of the vector quantization were used to train the models by the Forward-Backward algorithm. After performing the training experiments for each speaker, it was determined that each speaker had model parameters $\lambda_s = (A, B, \pi)$.

E. Classification Result

The evaluation of HMMs was performed by using the forward algorithm. The likelihood probability was computed for all models, and then we selected the HMM model with the highest likelihood.

1) Classification Rate

To determine the classification rate, this experiment was performed using all the dataset (i.e. a population of 40 speakers). An average classification rate was 80% for all the speakers.

The classification rates are summarized in Fig.8. The figure plots the classification rates for all the speakers. We notice that the classification rate of speaker#1 is 66% and classification rate of speaker#2, 3,…,39, 40 is %100. Speaker#22 has the lowest classification rate of 33%.

2) Population Size

This experiment was used to determine the effect of population size on identification performance. We used 9 independent datasets with following sizes: 3, 5, 10, 15, 20, 25, 30, 35, and 40 speakers. The result is shown in Fig. 9. This result indicates that as the size of the population increases, the identification rate decreases. This fact is a strong indicator that successful speaker identification cannot be performed on large populations, such as the population of an entire city, state or country.

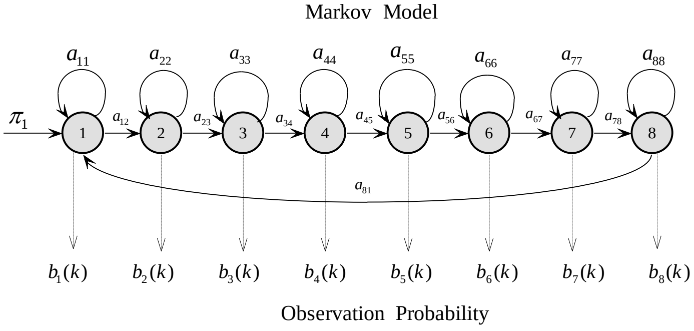

Figure 7. HMM-Model, λ , used for speaker identification.

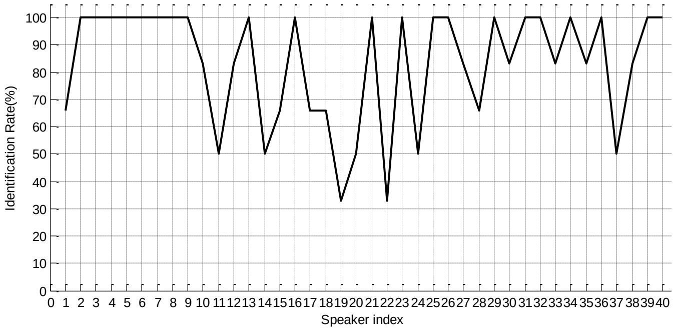

Figure 8. Classification results for all the speakers.

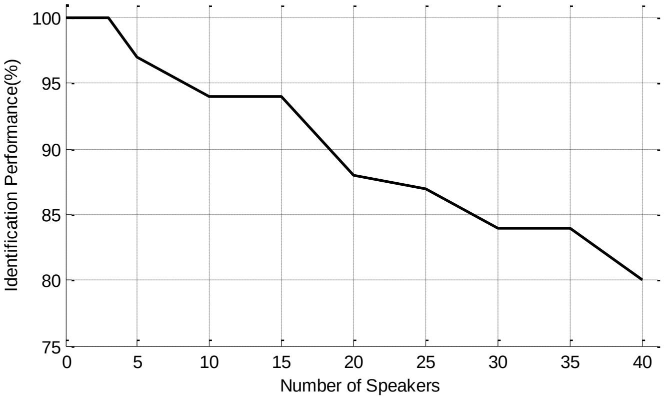

Figure 9. Identification rate as a function of the number of speakers.

3) Comparing HMM with LDA and MLP

This experiment was used to compare three classifiers: Linear Discriminant Analysis (LDA), Multilayer Perceptron Network (MLP) and Hidden Markov Model (HMM). Mel-Frequency Cepstrum Coefficients (MFCC) and Linear Predictive Coding Coefficients (LPCC) parameters were used to test the performance for each of the three classifiers. The experiment was performed using 3 speakers randomly selected from the dataset.

Table II shows the identification rates of the 3 speakers for the three classifiers. It can be seen that, the Hidden Markov Model (HMM) classifier outperforms the Linear Discriminant Analysis (LDA) and Multilayer Perceptron (MLP). It can also be seen that the MFCC gives higher identification rates than the LPCC.

TABLE II. IDENTIFICATION RESULTS USING LDA,MLP AND HMM

Vector Parameter	**Classifier**		
	Linear Discriminant Analysis	***Multilayer Perceptron***	***Hidden Markov Model***
MFCC	70.5%	82.1%	100%
LPCC	55.6%	75.1%	94%

V. CONCLUSIONS

In this paper, we have attempted to describe a Text-Independent Speaker Identification System based on MFCC feature vectors and HMMs recognizer.

We have extracted the feature vectors such as LPCC and MFCC. Then we have used the K-mean clustering algorithm to construct a codebook of length 128. Finally we have developed 40 Models. We have achieved identification rate of 80% using all the speakers in the dataset. Also we have tested the effect of population size on identification rate. The results have shown that, large population produces poor performance. Also we have compared between HMM, LDA and MLP. The experiments show that HMM classifier with MFCC feature vectors gives better classification rate.

VI. FUTURE RESEARCH

The identification rate achieved in this paper was carried out by using a close-set database. In the future research, we will apply an open-set database since an open-set may cause the experiment to be similar to real-life situations. We will study and use Gaussian Mixture Models (GMM) to estimate the probability function of the feature vectors. Future work will be focused on other models such as Support Vector Machine (SVM) classifier or Kernel method.

REFERENCES

[1] C. H. Lee, F. K. Soong and K. K. Paliwal, Automatic Speech and Speaker Recognition:Advanced Topics, Boston: Kluwer Academic Publishers, 1996.

[2] H. Beigi, Fundamentals of Speaker Recognition, New York: Springer Science and Business Media, Inc, 2011.

[3] L. R. Rabiner and R. W. Schafer, Digital Processing of Speech Signals, New Jersey: Prentice-Hall, 1978.

[4] R. L. Klevans and R. D. Rodman, Voice Recognition, Boston: Artech House, 1997.

[5] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of IEEE, vol. 77, no. 2, pp. 257-286, Feb. 1989.

[6] L.R.Rabiner and B. Juang, Fundamentals of Speech Recognition, New Jersey: Prentice-Hall, Inc, 1993.

[7] X. Huang, A. Acero and H. Hon, Spoken Language Processing: A Guide to Theory, Algorithm, and System Development, New Jersy: Prentice-Hall,Inc, 2001.

[8] L. R. Rabiner and R. W. Schafer, Theory and Applications of Digital Speech Processing, New Jersey: Pearson Higher Education, 2011.

[9] M. R. Schroeder, Computer Speech: Recognition, Compression, Synthesis, Berlin: Springer-Verlag., 2004.

[10] S. Chakroborty, A. Roy and G. Saha, "Improved Closed Set Text-Independent Speaker Identification by combining MFCC with Evidence from Flipped Filter Banks," International Journal of Information and Communication Engineering, Vol. 4, No. 2, pp.114-121, 2008.

[11] G. Ananthakrishnan, "Music and Speech Analysis Using the 'Bach' Scale Filter-Bank", M.S. thesis, Dept. Elec. Eng., Indian Institute of Science., Bangalore, India., April 2007.

[12] S. B. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," IEEE Transaction on Acoustics Speech and Signal Processing, vol. 28, no. 4, pp. 357-366, August 1980.

[13] G. A. Fink, Markov Models for Pattern Recognition: From Theory to Applications. Berlin: Springer-Verlag, 2008.

[14] W. Ching and M. K. Ng, Markov Chains: Models, Algorithms and Applications, New York: Springer Science+ Business Media, Inc, 2006.

[15] R. P. Agency, "Linguistic Data Consortuim," University of Pennsylvania, 19 Jul 2011. [Online]. Available: http://www.ldc.upenn.edu. [Accessed 9 May 2012].
.