Speech Recognition using MFCC and DTW

Bhadragiri Jagan Mohan[1], Ramesh Babu. N[1]
[1]School of Electrical Engineering
VIT University
Vellore, India
jagan968@gmail.com, nrameshbabu@vit.ac.in

***Abstract*— Speech recognition has wide range of applications in security systems, healthcare, telephony military, and equipment designed for handicapped. Speech is continuous varying signal. So, proper digital processing algorithm has to be selected for automatic speech recognition system. To obtain required information from the speech sample, features have to be extracted from it. For recognition purpose the feature are analyzed to make decisions. In this paper implementation of Speech recognition system in MATLAB environment is explained. Mel-Frequency Cepstral Coefficients (MFCC) and Dynamic Time Wrapping (DTW) are two algorithms adapted for feature extraction and pattern matching respectively. Results are obtained by one time training and continuous testing phases.**

Keywords*— Isolated word recognition; MFCC; DTW; FFT; Speech Recognition component

I. INTRODUCTION

Speech recognition algorithms can be broadly divided into speaker dependent and speaker independent. Speaker dependent system focuses on developing a system to recognize unique voiceprint of individuals. Speaker independent system involves identifying the word uttered by the speaker. It can be further classified into isolated word detection and continuous speech recognition [1]. Input for isolated word detection is single words separated by pauses. This is relatively simpler compared to continuous speech recognition as the system doesn't need to learn fluidic sequence of dictionary words. Continuous speech recognition can be used in security systems for verifying password uttered by the user. Speech recognition system processes the word uttered by the user and generates its features. When the user utters something, it is sent to the speech engine to be processed then converted into digital domain. The digitalized speech samples are processed to extract features using MFCC algorithm. Once the desired number of features is obtained, they can be sent through feature matching stage where DTW is used for comparison between saved templates and recorded speech. This entire system is implemented in MATLAB environment where the speech samples input are recorded through windows sound card [2,3].

II. LITERATURE REVIEW

Various methodologies have been proposed for isolated word detection and continuous speech recognition over the years [4]. Out of which Hidden Markov Models (HMM) has been extensively used in lot of speech recognition applications because of its high reliability [5]. Artificial Neural Networks (ANN) is another classifier of speech recognition with acceptable accuracy. Support Vector Machines (SVM) classifiers have been used to classify speech patterns using linear and non-linear discrimination models. For simple isolated word detection MFCC and DTW approach is enough and efficient [6]. However, if continuous speech detection with speaker discrimination is needed MFCC alone is not necessary for assuring the efficiency in the algorithm. Combination of various features is to be adapted in this case for high reliability [4]. Even for the implementation of speech recognition engine in embedded systems MFCC and DTW algorithms are proved to be simpler enough compared with neural networks and HMM [7].

III. METHODOLOGY

A. Recognition Module

Isolated word detection involves two digital signal processes which are Feature Extraction and Feature Matching. Feature extraction involves calculation of MFCCs for each frame. MFCCs are the coefficients that collectively represent the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear MEL scale of frequency. For feature matching DTW method is used.

B. Feature Extraction (MFCC)

MFCC is based on human hearing perceptions which cannot perceive frequencies over 1KHz. Features obtained by MFCC algorithm are similar to known variation of the human cochlea's critical bandwidth with frequency [6]. The process extracting MFCCs for a given voice sample is shown in Fig. 1

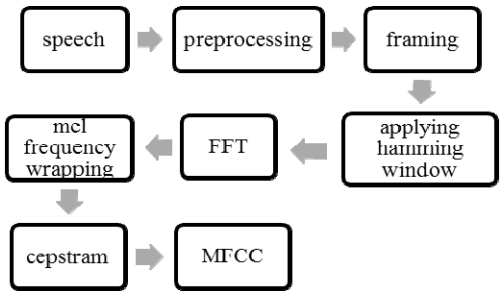

Fig. 1 MFCC flow diagram

Prior to feature extraction the voice sample has to undergo Analog to digital conversion followed by Pre-Emphasis and Filtering. It is important to have a sufficient sampling rate to avoid aliasing. According to the Nyquists sampling theorem the absolute minimum sampling frequency of a signal with maximum frequency *f* should be *2f* Hz. Digitalized voice sample for word 'FORWARD' is shown in Fig. 2.

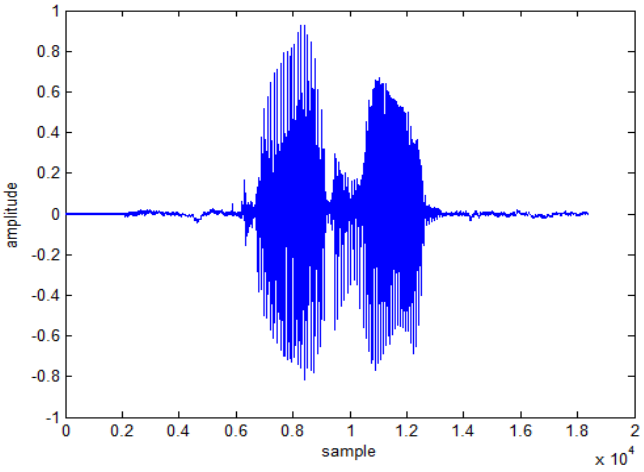

Fig. 2 Digitalized voice sample for word 'FORWARD'

Pre-emphasis stage increases the magnitude of higher frequency with respect to lower frequencies [8]. FIR filter, used for this purpose and its corresponding discrete output is given in equation 1 and equation 2 respectively.

$$F(z) = 1 - kz^{-1} \qquad 0 < k < 1 \qquad (1)$$

$$y[n] = s[n] - k.s[n-1] \qquad 0 < k < 1 \qquad (2)$$

Where, y[n] is the output and s[n] the signal input of the FIR filter.

The Noise-gate is applied to the pre-emphasized voice sample to remove the amplitudes (noise) below a particular threshold value. Spectral estimate of word 'FORWARD' after preprocessing and noise-gate is shown in Fig. 3.

After a noise-gate is applied the voice sample can be aligned to start from zero on the time axis. This is called zero-alignment. This can reduce some of the workload for the pattern matching process later in the program since the voice samples are much closer to each other than they otherwise would be. All the above signal operations are carried out prior to MFCC extraction. They avoid interaction of noise with significant features [8]. Zero aligned spectrum of word 'FORWARD' is shown in Fig. 4.

Voice sample is a time varying signal. It has to be framed with frame length within range of 20 ms to 30 ms. The frame length should not be too short such that we can obtain reliable spectral estimate for each frame. On the other hand it should not be to too long such that under a particular frame voice sample is time invariant [1,6]. Adjacent frames are being separated by M (M<N). In this paper frame overlapping is 100 and frame length is 256. Now each frame is multiplied with hamming window. The Hamming window function is expressed in (3). Output of each frame after filtering is obtained as in (4).

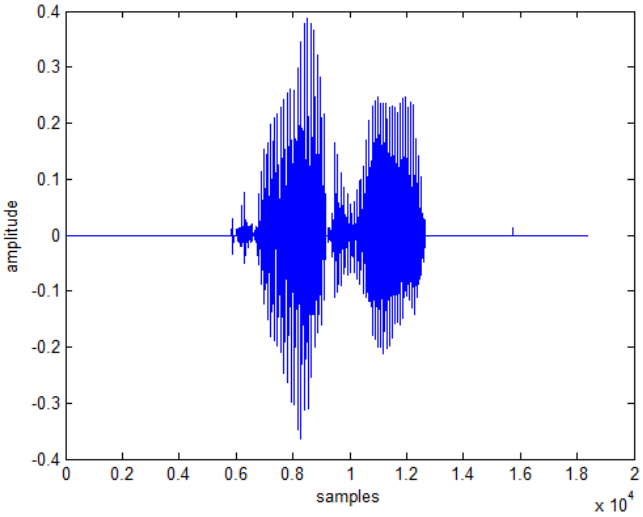

Fig. 3 spectral estimate of word 'FORWARD' after pre-processing and noise-gate.

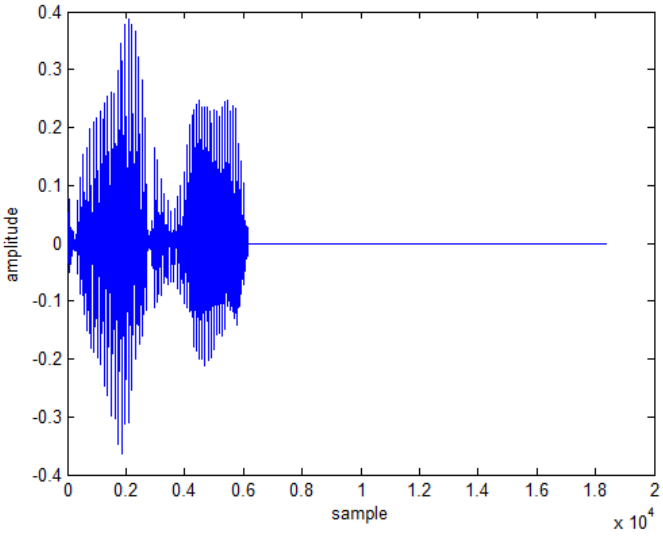

Fig.4 Zero aligned spectral estimate of word 'FORWARD'

$$W[n] = 0.54 - 0.46\cos\left[\frac{2\pi n}{N-1}\right] \qquad (3)$$

$$Y[n] = X[n] \times W[n] \qquad (4)$$

Where, N = number of samples in each frame
Y[n] = Output signal
X (n) = input signal
W[n] = nth coefficient of hamming window

Fast Fourier Transform (FFT) is applied to each frame which transforms signal to frequency domain. We would generally perform a 512 point FFT and keep only the first 257

coefficients. Thus the spectrum for each frame is obtained. But, it still contains lot of information not required for feature matching stage. The feature matching algorithm cannot discern the difference between two closely spaced frequencies [9]. For this reason we take clumps of spectral bins and sum them up to get an idea of how much energy exists in various frequency regions. This can be performed by multiplying each frame with Triangular MEL Filter banks shown in Fig. 5.

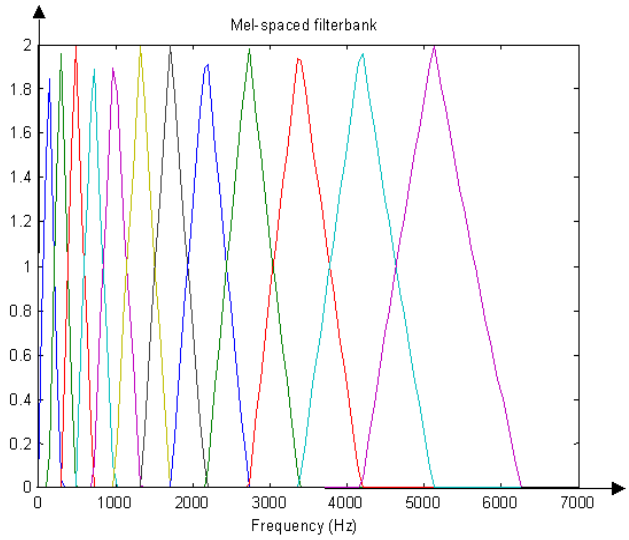

Fig. 5 Mel-spaced filter banks

The first filter is very narrow and gives us indication of how much energy exists near *zero* hertz. As the frequency gets higher our filters get wider as we become less concerned about variations. The equation for calculating MEL for a given frequency is shown in (5) [8].

$$F(MEL) = 2595 \times \log_{10}[1 + f/700] \qquad (5)$$

We are only interested in roughly how much energy occurs at each spot. Here a set of 26 triangular filters are taken. To calculate filter bank energies we multiply each filter bank with the energy spectrum, and then add up the coefficients. Once this is performed we are left with 26 numbers that give us an indication of how much energy was in each filter bank. Logarithm for these 26 energy values is taken following by Discrete Cosine Transform (DCT). DCT is calculated using equation shown in (6) [6].

$$C_n = \sum_{k=1}^{k} (\log S_k)\left[n\left(k - \frac{1}{2} \right)\frac{\pi}{K} \right] \qquad (6)$$

Where, n = 1, 2…K

S_k = FFT coefficients

Value of K is taken to be 26. Thus we are left with 26 coefficients but, for feature matching; only the lower 12-13 of the 26 coefficients are kept. The resulting features (12 numbers for each frame) are called Mel Frequency Cepstral Coefficients. Thus the sample which is in frequency domain after applying FFT is converted back to time domain using MEL filter and DCT as shown in Fig. 6.

Fig. 6. Steps in Converting from frequency domain to time domain

Then Delta and Delta-Delta coefficients are calculated for each frame. The first order derivative is called delta coefficient and the second order derivative is called delta-delta coefficient. The n^{th} Delta feature and Delta-Delta feature is then defined by (7) and (8).

$$\Delta f_k[n] = f_{k+M}[n] - f_{k-M}[n] \qquad (7)$$

$$\Delta^2 f_k[n] = \Delta f_{k+M}[n] - \Delta f_{k-M}[n] \qquad (8)$$

Where, M typically is 2-3 frames. The differentiation is done for each feature vector separately. Thus, for each frame we are left with 36 coefficients (12 MFCCs, 12 Delta, and 12 Delta-Delta).

C. *Feature Matching (DTW)*

In this stage, the features of word calculated in previous step are compared with reference templates. DTW algorithm is implemented to calculate least distance between features of word uttered and reference templates. Corresponding to least value among calculated scores with each template, the word is detected. DTW finds the optimal alignment between two times series if one time series may be "warped" non-linearly by stretching or shrinking it along its time axis [6]. The extent of matching between two time series is measured in terms of distance factor. Dynamic time wrapping for two voice samples is illustrated in Fig. 7.

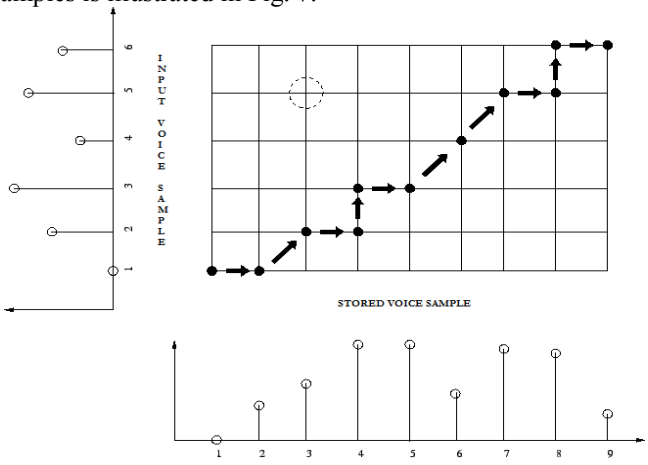

Fig. 7. Dynamic Time Wrapping of two voice samples

A matrix of order *n by m* is created whose *(i, j)* element is distance *d* (a_i, b_j) between points a_i and b_j of two time sequences. Euclidean computation is used to measure distance between features of input sample and saved template.
Then, distance is measured by (9).

$$D(i,j) = \min[D(i-1,j-1), D(i-1,j), D(i,j-1)] + d(i,j) \quad (9)$$

The template corresponding to least distance is the word detected.

IV. RESULTS AND DISCUSSION

The distances while comparing similar words and different words are shown in table1. With similar words, distance is below 100. With different words say, FORWARD and REVERSE distance is calculated to be more than 300 i.e., 334.11. Thus a Threshold of 150 or less can filter a given word from set of saved templates. As DTW calculates possible alignment between two vector paths, the distance obtained when two same sequences compared should be 0.

V. CONCLUSION

With MFCC and DTW, isolated word detection system is generated in MATLAB environment. System is trained by saving templates of five separate words. Results showed that saving ten templates for each word in training phase gives good results compared with five templates. Efficiency in detecting isolated words is 100 per cent for two syllable words compared with one syllable word. From the results above, we can infer that DTW distance between identical words is less than 100 and between different words is more than 290. So setting of threshold of 150 we can easily filter the word uttered by the user from the other words whose templates are saved in the training phase.

TABLE I
COMPARISON BETWEEN DIFFERENT WORDS

Word 1	Word 2	DTW distance
Forward	Left	382.4
Forward	Right	366.9
Forward	Reverse	334.11
Forward	Control	340.9
Left	Reverse	401.9
Left	Control	390.6
Left	Right	299.8
Right	Reverse	290.8
Right	Control	377
Control	Reverse	401.11

TABLE II
COMPARISION BETWEEN SAME WORDS

Similar words	DTW Distance
Forward	98
Left	76
Right	43
Reverse	77
Control	93

REFERENCES

[1] P.K. Sharma, B.R. Lakshmikantha and K.S. Sundar, "Real Time Control of DC Motor Drive using Speech Recognition", Proceedings of the 2010 India International Conference on Power Electronics (IICPE), Jan. 28-30, 2011, New Delhi, India, pp. 1-5.

[2] MFCC retrieved on Jan 23rd, 2013 from, http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

[3] Daryl Ning, "Developing an Isolated Word Recognition System in MATLAB", article retrieved from http://www.mathworks.in/company/newsletters/articles/developing-an-isolated-word-recognition-system-in-matlab.html.

[4] B.P. Das, R. Parekh, "Recognition of Isolated Words using Features based on LPC, MFCC, ZCR and STE, with Neural Network Classifiers", International Journal of Modern Engineering Research, Vol. 2, No. 3, June 2012, pp. 854-858.

[5] L. Rabiner, "A tutorial on Hidden Marcov Model and selected applications in Speech Recognition", Proceedings of the IEEE, Vol.77, No.2, 1989, pp. 257-286.

[6] L. Muda, M.Begam and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques", Journal of Computing, Vol. 2, No. 3, March 2010, pp. 138-143.

[7] M A Muqeet, "Speech Recognition using Digital Signal Proccessor", unpublished.

[8] A. Bala, Abhijit kumar, Nidhika Birla, "Voice Command Recognition System Based On MFCC And DTW", International Journal of Engineering Science and Technology, Vol. 2, No. 12, 2010, pp.7335-7342.

[9] M.R. Hasan, M. Jamil, M.G. Rabbani and M.S. Rahman, "Speaker Identification Using MEL Frequency Cepstral Coefficient", Proceedings of 3rd International conference on Electrical and Computer Engineering (ICECE), December,28-30, 2004, Dhaka, Bangladesh, pp. 565-568.