Response of Different Window Methods in Speech Recognition by Using Dynamic Programming

S. M. Shamsul Alam
Electronics and Communication Engineering Discipline
Khulna University, Khulna 9208
Bangladesh
alam_ece@yahoo.com

Sehrish Khan
Electronics and Communication Engineering Discipline
Khulna University, Khulna 9208
Bangladesh
pompey_01@yahoo.com

***Abstract*— In the area of pattern recognition, feature vectors identification is one of the major tasks to make this detection successful. For this reason, feature vectors depict the key distribution of data and by applying the various data analysis techniques, these vectors are obtained. In this paper, for identifying the good feature of speech signal, Mel Frequency Cepstral Coefficients (MFCC) are derived by applying the various window functions. After getting the feature vectors as MFCC, dynamic programming (DP) is used to get the distance for feature matching. In nutshell, to recognize individual speech, this paper presents the performances of various window techniques to make comparison between different voices by using (DP) algorithm. The responses of various types of window techniques including their energy spectrum are analyzed for extracting feature vectors from different speeches. Then this paper reports a DP based speech recognition technique. Result shows that for rectangular window based recognition, the feature extraction is very good and outperforms for word recognition in terms of low DP distance compared to other window methods.**

Keywords*— *Window Techniques; Dynamic Programming(DP); Speech Recognition; Log Spectrum; Mel Frequency Cepstral Coefficients (MFCC)

I. INTRODUCTION

Speech recognition refers to study of speech signals and its processing. This is one of the most important applications of signal processing concepts. Speech processing is a multidisciplinary field that involves linguistics, phonetics, human sound perception and acoustics. Current work summarizes the basic ideas and techniques used in speech processing.

Figure 1 shows the simplified model of a speech recognition system [1]. Here, voiced excitation is one of the major parts for speech generation technique. This is modeled by a pulse generator, which generates a pulse train (of triangle–shaped pulses) with its spectrum given as *P(f)*. In our voice generation system, there is one excitation known as the unvoiced excitation that is modeled by a white noise generator with spectrum *N(f)*. To mix voiced and unvoiced excitation, one can adjust the signal amplitude of the impulse generator (*v*) and the noise generator (*u*) [1]. Then these both spectrums are added and then passed through a spectral shaping filter of response *H(f)*. The emission characteristics of the lips is modeled by *R(f)*. Hence, the spectrum *S(f)* of the speech can be written as [1]:

$$S(f) = (v \cdot P(f) + u \cdot N(f)) \cdot H(f) \cdot R(f) = X(f) \cdot H(f) \cdot R(f) \quad (1)$$

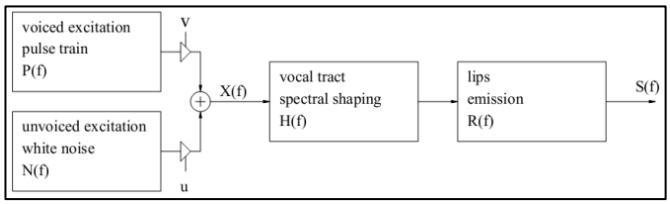

Fig. 1. A simple model of speech production [1].

From the above figure it can be shown that in power spectrum, it contains noise spectrum known as ripples which is caused by the excitation spectrum *X(f)*. So in acoustic processing, a special transformation is required to separate *X (f)* from *H (f)*.

In other words, it is required to eliminate the ripples by applying smooth spectral shape. For this reason most speech recognition systems use the so called Mel Frequency Cepstral Coefficients (MFCC). To identify the dynamic range, it is necessary to compute the speech parameters in short time intervals (like 10 ms). Therefore, frequency domain analysis (FFT) of this time domain speech signal is very important for recognizing or extracting features. For this reason, the speech signal is sampled and digitized to cut out a short piece of signal from the whole signal. This is done by multiplying the digitized signal with different window functions. In this paper, several window functions are used to identify the features of speech signals.

The response of window depends on its frequency domain behavior. For indentifying the proper window function, one preferred property is that, in the time domain analysis, the sum of window function ($w[n]$) with its shifted version by $M/2$ samples (M is the window order) would be constant [2]:

$$w[n] + w[n - M/2] = \text{constant} \quad (2)$$

For analyzing speech signals, those windows should be used which satisfy equation (2) because the cost computations is considerably decreased by using those windows [2]. The rectangular, Bartlett, Hann, and Hamming windows offer this advantage, but other windows such as Blackman, Kaiser, Gaussian, do not satisfy property of equation (2).

II. COMPUTATION OF SPEECH PARAMETERS

As we mentioned earlier that to identify the dynamic change, it is necessary to compute the speech parameters in short time intervals. Typically, the spectral parameters of speech are estimated in time intervals of 10ms. First it is required to sample and digitize the speech signal. Depending on the implementation, a sampling frequency f_s is required between 8 kHz and 16 kHz and usually a 16 bit quantization of the signal amplitude is used. After digitizing the analog speech signal, a series of speech samples $s(k\cdot\Delta t)$ is found, where $\Delta t = 1/f_s$. Now a pre emphasis filter is used to eliminate the -6dB per octave decay of the spectral energy:

$$\hat{s}(k) = s(k) - 0.97 \cdot s(k-1) \tag{3}$$

Here we have applied seven different types of window techniques and analyzed their results which will be discussed in simulation results section. In result and discussing section, I will discuss elaborately.

To compute the spectrum, DFT (Discrete Fourier Transform) is applied to convert the frequency domain transformation of short interval of speech signal. So the equation of this frequency domain transform is :

$$V(n) = \sum_{k=0}^{N-1} v(k) \cdot e^{-j2\pi kn/N} \; ; n = 0, 1, \ldots N\text{-}1 \tag{4}$$

Where

$$v(k) = \begin{cases} \hat{s}(k) \cdot w(k-m); & k = m, m+1 \ldots\ldots m+N-1 \\ 0 \; ; \text{else} \end{cases}$$

Here $\hat{s}(k)$ is the discrete time domain speech signal and $w(k)$ is window function depend on which type of window we will apply to get the spectrum of short sample of speech signal. Power spectrum is found by squaring the magnitude $|V(n)|$.

A. Mel Spectral Coefficients

It is experimentally proved that human ear does not show a linear frequency resolution but builds several group of frequencies and integrate the spectral energies within a given group. Furthermore, the mid-frequency and bandwidth of these groups are non–linearly distributed. The non–linear warping of the frequency axis can be modeled by the so–called mel-scale. The frequency groups are assumed to be linearly distributed along the mel-scale. The so called mel–frequency f_{mel} can be computed from the frequency f as follows [1]:

$$f_{mel}(f) = 2595 \cdot \log\left(1 + \frac{f}{700Hz}\right) \tag{5}$$

If we plot the equation (5) figure 2 is found. From this figure, human ear has high frequency response in low–frequency parts of the spectrum and low frequency response in the high–frequency parts of the spectrum.

As we mentioned earlier that $V(n)$ is the frequency domain transformation of short piece of speech. Now the power spectrum coefficients $|V(n)|^2$ are transformed to reflect the frequency resolution of human ear. Ḱ triangle shaped window is applied in the spectral domain to build a weighted sum over those power spectrum coefficient $|V(n)|^2$ which lie within the window. Ḱ triangle shaped window coefficients are denoted as below

$$\eta_{kn}; k = 0,1 \ldots\ldots \text{ Ḱ} \; ; n = 0, 1, \ldots\ldots N/2.$$

So the mel spectral coefficients of voice signal is

$$G(k) = \sum_{n=0}^{N/2} \eta_{kn} \cdot |V(n)|^2 \; ; \text{k} = 0, 1, \ldots\ldots \text{ Ḱ -1} \tag{6}$$

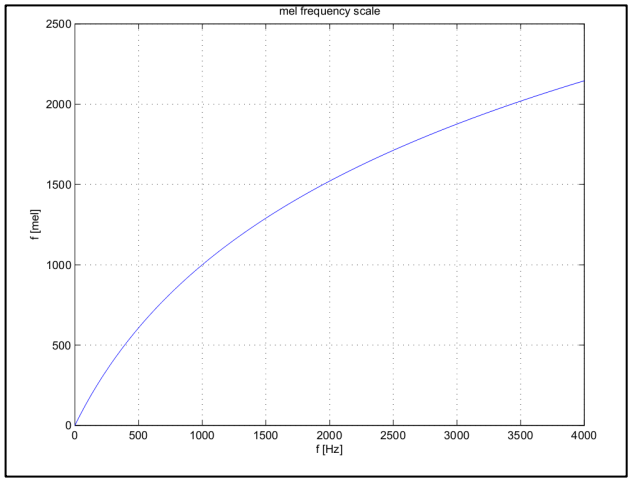

Fig. 2. Frequency measured in mel frequency scale [1].

B. Cepstral Transformation

The spectrum of the speech signal is described by equation (1). Since the transmission function of vocal tract $H(f)$ is multiplied with the spectrum of the excitation signal $X(f)$, the unwanted ripples are included in the spectrum. But in speech recognition task smooth spectrum is required which should represent only $H(f)$ but not $X(f)$. To cope with this problem, cepstral analysis is used. From equation (1), it can be shown that the product of spectral functions into the interesting vocal tract spectrum is separated from the part describing the excitation and emission properties. By applying the logarithmic function to both side of equation (1), we get:

$$\log(S(f)) = \log(H(f) \cdot U(f)) = \log(H(f)) + \log(U(f)) \tag{7}$$

Now taking the square of absolute value of power spectrum equation (7) and apply logarithmic function:

$$\log(|S(f)|^2) = \log(|H(f)|^2) + \log(|U(f)|^2) \tag{8}$$

Equation (8) shows the log power spectrum of speech which contains the unwanted ripples signal $U(f)$. In log-spectral domain, the spectrum of $U(f)$ can be subtracted if it is known properly. But it is very difficult to know the exact information

of this *U(f)* signal. But if we convert or transform this log spectrum to time domain then this ripple spectrum allocates the high frequency region then it is very easy to put them zero. Therefore, this sort of sense is known as low pass filtering techniques. So, based on this theory the following equation is formed:

$$\hat{s}(d) = FT^{-1}\{\log(|S(f)|^2)\} = FT^{-1}\{\log(|H(f)|^2)\} + FT^{-1}\{\log(|U(f)|^2)\} \quad (9)$$

Applying the inverse DFT to the log power spectrum coefficients $\log(|V(n)|^2)$, the following equation is formed:

$$\hat{s}(d) = \frac{1}{N}\sum_{n=0}^{N-1}\log(|V(n)|^2 \cdot e^{j2\pi dn/N}); d = 0, 1,\ldots N\text{-}1 \quad (10)$$

Now the low pass filtering of energy spectrum can be done by setting the higher valued coefficients of $\hat{s}(d)$ to zero and then transforming back into the frequency domain. This process is known as liftering. This is the smoothed version of log power spectrum [1].

Figure 3 shows the effect of this liftering effect in real time voice 'Hello' [1]. So we see that after liftering its response is very good in terms of smoothness. It can be said that after applying smoothing technique it contains less noise than previous condition. For that reason the SNR value of this smooth signal is higher than no other smoothing signal. If we deduct these two signals then we'll get rough picture of SNR value.

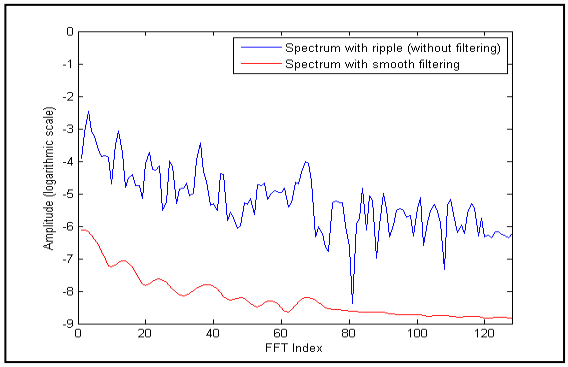

Fig. 3. Log power spectrum of smoothing and non smoothing filtering [1].

C. Mel Frequency Cepstral Coefficients (MFCC)

Here we will show how to compute the mel cepstrum used in speech recognition. As stated above, for speech recognition, the mel spectrum is used to reflect the perception characteristics of the human ear. In analogy to compute the cepstrum, it is required to take the logarithm of the mel power spectrum and transform into the frequency domain to compute the mel cepstrum. The Q coefficients of the mel cepstrum are used in typical speech recognition system which is also known as feature vectors. The required mathematics for calculating this feature vectors are given below. Here DCT is applied for calculating MFCC [1].

$$c(q) = \sum_{k=0}^{K-1}\log(G(k))\cdot\cos\left(\frac{\pi q(2k+1)}{2K}\right) \quad (11)$$

where $q = 0, 1 \ldots . Q-1$

Here equation 11 shows the DCT of *G(k)* which is defined as mel spectral coefficients of voice signal . While successive coefficients *G(k)* of the mel power spectrum are correlated, the MFCCs resulting from the cosine transform are decorrelated. The MFCC are used directly for further processing in the speech recognition system instead of transforming them back to the frequency domain.

III. Feature Matching Algorithm

Speech signal is represented by a series of feature vectors which are computed every short interval (for example 10ms). For an utterance of a word *w* which is *Tx* vectors long a sequence of vectors $\tilde{X} = \{\vec{x}_0, \vec{x}_1 \ldots . \vec{x}_{T_x-1}\}$ is generated from the acoustic processing stage. Dynamic programming (DP) is a way to compute a "distance" between unknown sequence of $\tilde{X}$ and known sequence of vectors $\tilde{W} = \{\vec{w}_0, \vec{w}_1 \ldots . \vec{w}_{kT_{wk}}\}$ [6]. Suppose $D(\tilde{X}, \tilde{W}_{k,\omega_v})$ is a distance between $\tilde{X}$ and a prototype $\tilde{W}_{k,\omega v}$, so the class distance $D_{\omega_v}(\tilde{X})$ is defined as follows:

$$D_{\omega_v}(\tilde{X}) = \min_k\{D(\tilde{X}, \tilde{W}_{k,\omega_v})\}; k = 0,1 \ldots ., (K_{\omega_v} - 1)$$

where ω_v is the class of utterances whose orthographic representation is given by the word w_v [3]. Then the classification task is as follows[1]:

$$\tilde{X} \in \omega_v \Leftrightarrow v = \arg\min_v\{D_{\omega_v}(\tilde{X})\} \quad (12)$$

This definition implies that this recognizer has a set of vocabulary classes denoted by $\Omega = \{\omega_0, \omega_1, \omega_2, \ldots . \omega_{(v-1)}\}$. The only criterion the classifier applies for its choice is the acoustic similarity between the utterance $\tilde{X}$ and the known prototypes $\tilde{W}_{k,\omega_v}$ as defined by the distance measured as $D(\tilde{X}, \tilde{W}_{k,\omega_v})$. So to measure this minimum distance, proper identification of features is one of the major tasks of speech detection. After extracting the features from different speech, DP is allowed to compare two sequences of vectors in terms of $DP_{distance.}$

A. Finding the Optimal Path (Dynamic Time Warping)

The criteria for optimal path is denoted by the following equation:

$$P_{opt} = \arg\min_p\{D(\tilde{X}, \tilde{W}, P)\} \quad (13)$$

It may not be necessary to compute all possible paths P corresponding distances $D(\widetilde{X},\widetilde{W},P)$ to find the optimum distance. As it is known that both sequences of vectors represent feature vectors measured in short time intervals. The reasonable boundaries are set as: the first vector of $\widetilde{X}$ and $\widetilde{W}$ should be assigned to each other as their last vectors [1]. For the time indices in between one should avoid the giant leap backward or forward in time, but try to "reuse" of the preceding vector(s) to locally warp the duration of a short segment of speech signal [1]. With these restrictions, it can be possible to draw a diagram of possible "local" path alternatives for one grid point and its possible predecessors (of course, many other local path diagrams are possible).

Figure 4 shows the predecessor paths for a given grid point. All possible paths *P* that will be considered as possible candidates for being the optimal path P_{opt} can be constructed as a concatenation of the local path. To reach a given grid point (*i,j*) from (*i* -1, *j*-1), the diagonal transition involves only the single vector distance at grid point (*i,j*) as opposed to using the vertical or horizontal transition, where also the distances for the grid points (*i* -1, *j*) or (*i*, *j*-1) would have to be added. To compensate this effect, the local distance $d(\vec{W}_i,\vec{X}_j)$ is added twice when using the diagonal transition.

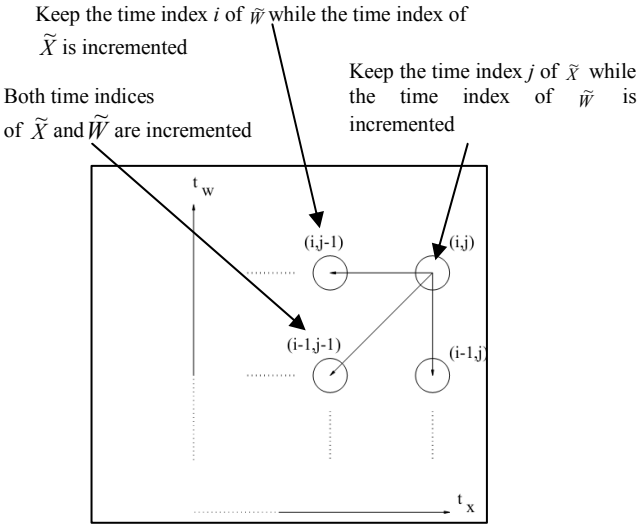

Fig. 4. Local path alternatives for grid point [1].

Using Bellman's Principle, the optimal path can be found by reversely following all the local decisions down to the origin (0,0). This procedure is called backtracking which is known as Dynamic Programming (DP) or Dynamic Time Warping (DTW) [1].

IV. METHODOLOGY

In previous sections, we have developed how to extract the feature vectors of speech signals and how to compare two feature vectors. By using these two explanations, we have showed our research works here. As we mentioned earlier that window techniques are applied for extracting the feature vectors. So we have applied several window functions to get these feature vectors. Here **Hamming, Hann**, **Rectangular**, **Kaiser**, **Blackman**, **Bartlett** and **Gaussian** windows are taken for consideration. The response of window depends on its frequency domain behavior. It is mentioned earlier that equation (2) is one preferred property for selecting window in data spectrum analysis. That means in the time domain, the sum of window function (*w[n]*) with its shifted version by *M*/2 samples (*M* is the window order) would be constant [2].

The properties of the different window functions are given as below [5]:

Rectangular window

$$w[n]=\begin{cases}1, & 0\le n\le M,\\ 0, & otherwise\end{cases}$$

Hamming window

$$w[n]=\begin{cases}0.54-0.46\cos(2\pi n/M), & 0\le n\le M,\\ 0, & otherwise\end{cases}$$

Hann window

$$w[n]=\begin{cases}0.5-0.5\cos(2\pi n/M), & 0\le n\le M,\\ 0, & otherwise\end{cases}$$

Blackman window

$$w[n]=\begin{cases}0.42-0.5\cos\left(\frac{2\pi n}{M}\right)+0.08\cos(4\pi n/M), & 0\le n\le M,\\ 0, & otherwise\end{cases}$$

Bartlett window

$$w[n]=\begin{cases}\frac{2n}{M}, & 0\le n\le \frac{M}{2}, \quad M \quad even\\ 2-\frac{2M}{n}, & \frac{M}{2}<n\le M\\ 0, & otherwise\end{cases}$$

Kaiser window

$$w[n]=\begin{cases}\frac{I_o\left[\beta\left(1-\left[(n-\alpha)/\alpha\right]^2\right)^{1/2}\right]}{I_o(\beta)}, & 0\le n\le M,\\ 0, & otherwise\end{cases}$$

Gaussian window

$$w[n]=\begin{cases}e^{-\frac{1}{2}\left(\alpha\frac{n}{M/2}\right)^2}, & -\frac{M}{2}\le n\le\frac{M}{2},\\ 0, & otherwise\end{cases}$$

For analyzing speech signals, preferred windows should satisfy the equation (2). In analyzing the non-stationary signals like speech processing, the signal is partitioned into several overlapped frames to yield near stationary properties. Therefore, these frames are the windowed version of the

original signal. However, the rectangular, Bartlett, Hann, and Hamming windows offer this advantage, but other windows such as Blackman, Kaiser, Gaussian, do not satisfy the property of (2).

The block diagram of feature extraction is given in figure 5.

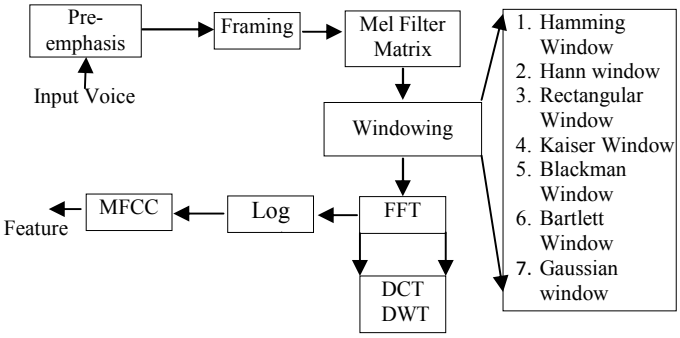

Fig. 5. Block diagram of feature extraction techniques for input voice signal.

Figure 5 represents the basic block diagram of feature extraction technique in terms of MFCC. Here pre-emphasis and framing blocks are related to input filtering, sampling frequency, time and no of channels. The mathematical derivation and definitions of other blocks are already discussed in previous sections. Under the windowing, one of the window is selected and we have taken the listed windows and compared their responses under log spectrum analysis.

Figure 6 shows the system block diagram of recognition techniques. Figure 6 represents the speech recognition technique. Here first the feature vectors of different voices are stored in database then an arbitrary voice is applied for recognition. After extracting the feature of this arbitrary speech, DTW compares this arbitrary feature vector with stored feature vectors and gives the distance of each measurement. Based on this distance, the minimum value of distance identified as recognized word.

So the performance or success rate depends on the generation of feature vectors and pattern comparator technique. The figure 7 shows the flowchart of Automatic Speech Recognizer (ASR).

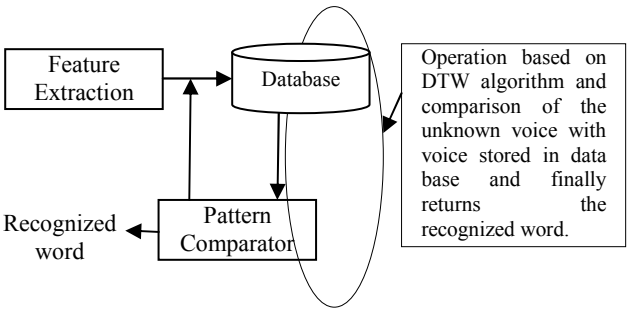

Fig. 6. Basic recognition technique.

V. SIMULATION RESULTS

The time domain behavior of the voice signal of word "Hello" is shown in Figure 8. It shows the sample waveform of our four speech vectors. Besides "Hello" we have taken other three words to enrich our database. From this speech vectors, the feature vector of each voice signal is identified as 22 x 147 dimension matrix. This feature vector for each given signal is stored in database.

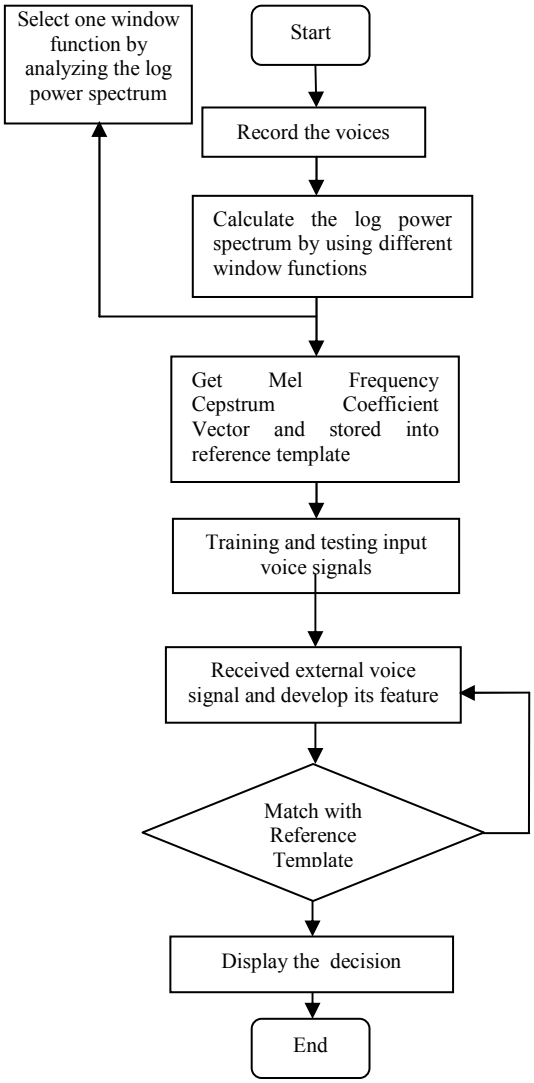

Fig. 7. Flowchart of ASR(Automatic Speech recognizer).

The responses of Different window functions are analyzed for real speech word "Hello". Here Hamming, Hann, Rectangular, Kaiser, Blackman, Bartlett and Gaussian windows are taken for consideration and figure 9 shows the log power spectrum of all these window functions.

From figure 9, it can be shown that rectangular and Kaiser window show very good performance in terms of power calculations. Besides this, these window functions represent better smoothing behavior than other window functions. So, rectangular window function can be used for feature extraction in speech recognition. Table I shows the comparison result of different window functions in terms of $DP_{distance}$. Here four different voices are taken for calculating the distance between different feature vectors i.e. $D(\widetilde{X}, \widetilde{W}_{k,\omega_v})$.

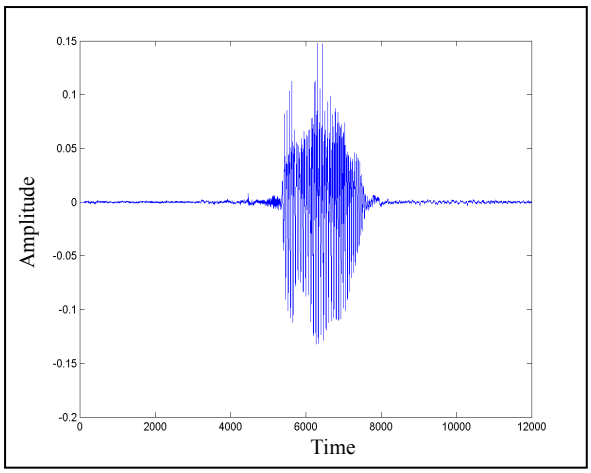

Fig. 8. Real time voice signal of the word /Hello/(Fs 8 KHz).

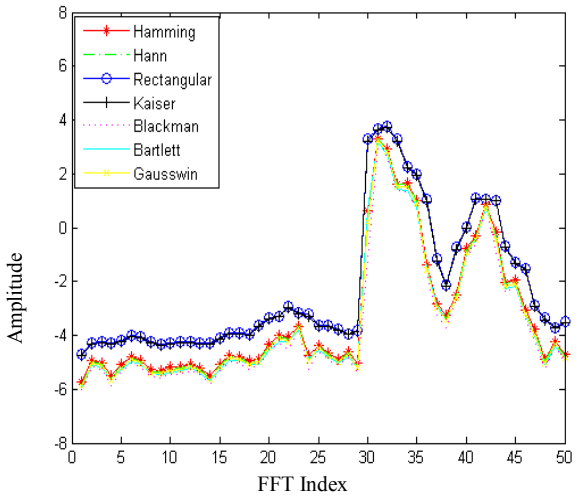

Fig. 9. Log power spectrum of the word /Hello/ (fs = 8 kHz, N = 256).

TABLE I. RESPONSE OF DIFFERENT WINDOW FUNCTIONS IN TERMS OF DYNAMIC PROGRAMMING DISTANCE

Name of window	$DP_{distance}$ (Dynamic Programming distance)			
Window Function	Speech 1	Speech 2	Speech 3	Speech 4
Hamming	503.0783	606.4834	481.5268	558.8639
Hann	515.0137	613.8545	490.7390	568.1450
Rectangular	**419.3535**	**522.4511**	**443.5564**	**501.7295**
Kaiser	419.8803	523.4796	442.3071	501.9276
Blackman	543.5555	636.8195	509.1822	584.2372
Bartlett	499.2801	601.6781	476.4476	556.2980
Gaussian	516.4941	617.3757	489.9639	566.0629

As it is discussed, earlier that DP is used to measure the distance of two sequences of vectors. To generate the sequence of vectors, speech signal is necessary to convert in discrete values. Thus good feature vectors are obtained by slicing those sequences using different window functions. DP takes two different feature victors to identify or recognize the matching between known and unknown sequences. Rectangular window satisfies the condition of equation (2) and it also shows the good frequency response of speech signal. In table I the response of rectangular window is good compared to other windows in terms of measuring feature distances. Figure 10 shows the optimal assignments of two speech vectors which is generated by two different voices. Here DP is used to get this optimal path. The success rate is almost equal to 98% which is measured by experimental approach.

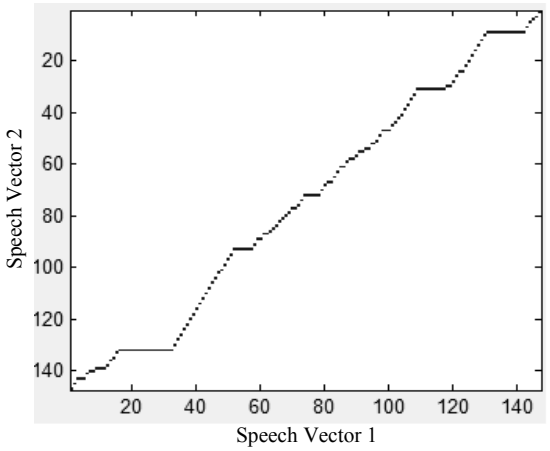

Fig. 10. Assignments between speech vector pairs.

In this paper, we have taken four different words as sample signals. For each signal we have calculated its feature vector by applying seven window functions and these vectors are stored in database. After that, for recognition we have taken one test word for calculating its feature vector and measuring the distance by using DP algorithm. The lowest distance shows the best match between test and stored vectors. Table I depicts such distance for using different kinds of windows.

VI. CONCLUSION

Feature extraction by using different window methods and distance measurement of two vectors are discussed extensively in this paper. This concept is widely used in efficient automatic voice recognition system. So to design effective speech processing and recognition system, this result may give sufficient information to select window function.

REFERENCES

[1] B. Planner, “An introduction to Speech recognition,” Tech. rep., University of Munich Germany, 2005.

[2] M. M. Kashtiban, M. G. Shayesteh, “A New Window Function for Signal Spectrum Analysis and FIR Filter Design,” 18th Iranian Conference on Electrical Engineering (ICEE) Iran, pp. 215-219, may 2010.

[3] J. R. Deller, J. L. Hansen, J. G. Proakis,”Discrete-Time Processing of Speech Signals”, John Wiley & Sons, 2000

[4] E. G. Schukat-Talamazzini. Automatische Spracherkennung. Vieweg Verlag, 1995.

[5] J. G. Proakis, D. G. Manolakis, “Digital Signal Processing Principles, Algorithms, and Applications”, Prentice-Hall Inc, 2007.

[6] H. Ney. “The use of a one-stage dynamic programming algorithm for connected word recognition”, IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP-32(2):263–271, April 1984.