ISSN No:-2456-2165

The Algorithms of Speech Recognition,Programming and Simulating in Matlab

Author: Mantu Bera
University Roll No:10511214008

A Thesis presented in partial fulfillment for the degree of
Masters of Technology in Computer Science & Engineering

Department Computer Science & Engineering
Bankura Unnayani Institute of Engineering
Bankura - 722 146, West Bengal, India
MAY 2016

ISSN No:-2456-2165

Dedicated to my parents

"We are what our thoughts have made us; so take care about what you think. Words are secondary. Thoughts live; they travel far."

- Swami Vivekananda

"Take up one idea. Make that one idea your life - think of it, dream of it, live on that idea. Let the brain, muscles, nerves, every part of your body, be full of that idea, and just leave every other idea alone. This is the way to success."

- Swami Vivekananda

"When an idea exclusively occupies the mind, it is transformed into an actual physical or mental state."

- Swami Vivekananda

"You have to dream before your dreams can come true."

-Dr. A. P. J. Abdul Kalam

"Dream is not that which you see while sleeping it is something that does not let you sleep."

-Dr. A. P. J. Abdul Kalam

ISSN No:-2456-2165

Acknowledgments

I hereby wish to express my sincere gratitude and respect to Assistant Prof. Tapas Sangiri, Dept. of CSE, Bankura Unnayani Institute Of Engineering, Bankura under whom I had proud privilege to work. His valuable guidance and encouragement have really led me to the path of completion of this project. Any amount of thanks would not be enough for the valuable guidance of my supervisor. I would also like to thank all the faculty member of CSE dept. for their devoted help. I also cordially thank all laboratory assistants for their cooperation. Finally, I would like to pen down my gratitude towards my family members for their continuous support and encouragement. It would have not been possible to complete my work without their support.

ISSN No:-2456-2165

ABSTRACT

My aim of this thesis work is to Test the algorithms of speech recognition. I have programmed and simulated the designed systems for speech recognition's Algorithms in MATLAB.Hence,There are two systems have been designed within this thesis.First One is based on the shape information of the cross-correlation plotting. The Second one is to use the Wiener Filter for realizing the speech recognition. The simulations of the systems which have been Programmed in MATLAB are accomplished by using the microphone for recording of the speaking words. After running the program in MATLAB, The system will ask people to record the words for three times.First and second recorded words are different words which will be used as the reference signals in the designed systems. The third recorded word is the same word as the one of the first two recorded words. After recording completed, the words will become the signals' information that will be sampled and stored in MATLAB. Then MATLAB will be able to give the judgment that which word has been recorded at the third time comparing with the first two reference words according to the algorithms programmed in MATLAB. I invite different people from different Places with different utterence to test the designed systems. The results of simulations for both designed systems prove that both of the designed systems work well whenever the first two reference recordings and the third time recording are recorded from the same person. But the designed systems have the defects when the first two reference recordings and the third recording are recorded from the different people. However, if the testing environment is quiet enough and the speaker is the same person for recordings of three times, the successful probability of the speech recognition approach to 100 percent. Thus, the designed systems really work well for the speech recognition.

Keywords:-Algorithm of Speech recognition, MATLAB programming, Recordings, Cross-correlation and auto correlation,FIR Wiener Filter, Simulations.

ISSN No:-2456-2165

CHAPTER-1
INTRODUCTION

The Speech recognition is a popular and significant topic in today's life.Applications of Speech recognition may be found in everywhere, that make our life more effective. For example the applications in the mobile phone, instead of typing the name of a person who want to call, people can just directly speak the name of the person to the mobile phone, and the mobile phone shall automatically call that person. If people want to send some text messages to anyone, people may also speak messages to the mobile phone instead of typing. Speech recognition is a technology that people can control the system with their speech or voice command. Instead of typing the keyboard or operating the buttons for the system,By using speech to control system is more convenient. It can also reduce the cost of the industry production at the same time. Use the speech recognition system not only improves the efficiency of the daily life, but also makes people's life more diversified.

Generally, the objectives of this thesis is to investigate the speech recognition's Algorithm by programming and simulating the designed system in MATLAB.The other purpose of this thesis is to utilize the learnt knowledge to the real applications.In this thesis, I will program two systems. The main algorithms for these two designed systems are cross-correlation and FIR Wiener Filter. To observe if these two algorithms can work for the speech recognition, I will invite different people from different Places to test the designed systems. In order to get reliable results, the tests will be completed in different situations. First, the test environments will be noisy and noiseless respectively to investigate the immunity of the noise for the designed systems.The test words will be chosen as different pairs that are the easily recognized words and the difficulty recognized words. As the two designed systems needs three input speech words that are two reference speech words and one target speech word, so it is significant to check if the two designed systems work well when the reference speech words and the target speech words recorded from the different person.

CHAPTER-2
BACKGROUND THEORY

2.1 Mathematical Equations

The theory part includes some information and definitions which will be involved in this thesis. I need this compulsory information to support my own research.Concerning and utilizing the theoretic knowledge,I have achieved my aim of this thesis. Including DC level and sampling theory, FFT, DFT, spectrum normalization, the algorithm of Cross-Correlation, the autocorrelation algorithm, FIR Wiener Filter,and use of spectrogram function to get the desired signals.

2.2 Sampling Theory:

In this thesis, I will use the microphone to record the person's analog speech signal through the computer, so the data quality of the speech signal will decide the quality of the speech recognition.The sampling frequency is one of the decisive factors for the data quality.In General, the analog signal can be written as:

$$x(t) = \sum_{i=1}^{N} A_i Cos(2\pi f_i t + \phi_i) \qquad (2.1)$$

The analog signal consists of different frequencies' components. Let, there is only one frequency component in this analog signal, and there is no phase shift. So this analog signal becomes:

$$x(t) = Acos(2\pi ft) \qquad (2.2)$$

We Know that the analog signal cannot be directly applied to the computer. It is essential to sample the analog signal x (t) into the discrete-time signal x (n)which is a collection or set of N samples 0 to(N-1), that the computer can use to process. In General, the discrete signal x (n) is always regarded as one signal sequence or a vector. So MATLAB can do the computation for the discrete-time signal.Time period of the analog signal x (t) is T. The sampling period of the discrete-time signal is TS. I assume that the analog signal is sampled from the initial time 0,As known, the relation between the analog signal frequency and time period is reciprocal. So the sampling frequency of the sampled signal is fs=1/Ts. Let, the length of x (n) is N for K original time periods. So, the relation between T and Ts is N×Ts=K×T. So N/K= T/Ts=fs/f, where both N and K are integers.If this analog signal is exactly sampled with the same sampling space and also the sampled signal is periodic, then N/K is integer also. Otherwise, the sampled signal will be aperiodic.

2.3 Time Domain To Frequency Domain:Dft And Fft

- DFT:

The DFT is just only a type of Fourier Transform for the discrete-time x (n) instead of the continuous analog signal x (t). The Fourier Transform equation is given below:

$$x(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{j\omega n} \qquad (2.3)$$

From the above equation, the main function of the Fourier Transform is to transform the variable from the variable n into the variable ω

That means transforming the signals from the time domain into the frequency domain.Assume that the recorded voice signal x(n) is a sequence or vector that consists of complex values, such as x(n)=R+I, where R is the real part of the value, and I is the imaginary part of the value. Since the exponent factor is:

$$e^{j\omega n} = cos(\omega n) + j.sin(\omega n) \tag{2.4}$$

$$X(n).e^{j\omega n} = (R+I).[cos(\omega n) + j.sin(\omega n)] = R.cos(\omega n) + R.j.sin(wn) + I.cos(\omega n) + I.sin(\omega n) \tag{2.5}$$

Rearranging the real part and image part of the equation we will get:

$$X(n).e^{j\omega n} = [R.cos(\omega n) + I.cos(\omega n)] + [R.j.sin(\omega n) + I.j.sin(\omega n)] \tag{2.6}$$

So,the equation(2.3) becomes:

$$x(\omega) = \sum [Rcos(\omega n) + Icos(\omega n)] + \sum [Rsin(\omega n) + Isin(\omega n)] \tag{2.7}$$

The equation (2.7) is also made of the real part and the imaginary part. Hence, in general situation, the real value of the signal x (n) is used.If the imaginary part I=0. Then the Fourier Transform is:

$$x(\omega) \sum_{n=-\infty}^{\infty} [Rcos(\omega n)] + \sum_{n=-\infty}^{\infty} [\, jRsin(\omega n)] \tag{2.8}$$

Assume that the frequency ω is set in $[0,2\omega]$, $X(\omega)$ may be regarded as an integral or the summation signal of all the frequency components. So ,the frequency component X (k) of $X(\omega)$ is got by sampling the entire frequency interval $\omega = [0,2\pi]$ by N samples. So it means the frequency component $\omega_k = k*(2\pi/N)$ And the DFT equation for the frequency component ω_k is as below:

$$X(k) = X(\omega_k) = \sum_{n=-\infty}^{\infty} x(n) e^{j\omega_k n} = \sum_{n=0}^{N-1} x(n).e^{j.(2\pi k/N)n} \tag{2.9}$$

This equation used to calculate the magnitude of the frequency component. The key of understanding DFT is to sample the frequency domain.

MATLAB are dealing with the data for vectors and matrixes. Definitely, understanding the linear algebra or matrix process of the DFT is important.After observing the equation (2.3), except the summation operator, the equation consists of 3 parts: output $X(\omega)$, input x (n) and the phase factor $e^{j\omega_k n}$. Since all the information of the frequency components is from the phase factor $e^{j\omega_k n}$.Then the phase factor can be denoted as:

$$W_N^{kn} = e^{jw_k n} \tag{2.10}$$

$$W_N^{kn} = e^{jw_k n} = [W_N^{0k}, W_N^{1k}, W_N^{2k}, ..., W_N^{(N-1)k}] \tag{2.11}$$

$$x(n) = [x(0), x(1), x(2), ...x(N-1)] \tag{2.12}$$

So,the equation(2.9)of the frequency component X(k)is just the linear product of the $(W_N^{kn})^H$ $and x(n)$:

$$x(k) = (W_N^{kn})^H .x(n) \quad (2.13)$$

It is the vector form about calculating frequency component by using DFT method. But if the signal is a actually long sequence, and the memory space is finite, then the using DFT to get the transformed signal will be limited. The faster and more efficient computation of DFT is FFT.I will introduce briefly about FFT in next section.

- *FFT:*

The FFT is the Fast Fourier Transform. Essentially, the FFT is still the DFT to transform the discrete-time signal from time domain into its frequency domain. The only difference is that the FFT is faster and more efficient on computation. And there are:

$$(k) = \sum_{n=0}^{n-1} x(n) W_N^{kn}, (k = 0,1,2,3....., N-1) \quad (2.14)$$

First, separate x(n) into two parts: x(odd)=x(2m+1) and x(even)=x(2m), where m=0, 1, 2,…,N/2-1.So, the N-point DFT equation also becomes two parts for each N/2 points:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} = \sum_{m=0}^{N/2-1} x(2m) W_N^{2mk} + \sum_{m=0}^{N/2-1} x(2m+1) W_N^{(2m+1)k} \quad (2.15)$$

$$= \sum_{m=0}^{N/2-1} x(2m) W_N^{(2m+1)k} + W_N^k \sum_{m=0}^{N/2-1} x(2m+1) W_N^{2mk}$$

$$e^{j\omega_k n} = cos(\omega_k n) + j.sin(\omega_k n) \quad (2.16)$$

$$e^{j(\omega_k+\pi)n} = cos[(\omega_k+\pi)n] + j.sin[(\omega_k+\pi)n] = -e^{j\omega_k n} \quad (2.17)$$

$$e^{j(\omega_k+\pi)n} = e^{-j\omega_k n} \quad (2.18)$$

When the phase factor is shifted with half period, then the value of the phase factor will not be changed, but the sign of the phase factor will be opposite(negative).That is called symmetry property of the phase factor. Since the phase factor can be also expressed as $W_N^{kn} = e^{j\omega_k n}$

$$W_N^{(k+N/2)n} = -W_N^{kn} \quad (2.19)$$

$$W_N^{kn})^2 = -W_{N/2}^{kn} = e^{j(4\pi k/N)n} \quad (2.20)$$

The N-point DFT equation finally becomes:

$$X(k) = \sum_{m-0}^{N/2-1} x_1(m) W_{N/2}^{mk} + W_N^k \sum_{m=0}^{N/2-1} x_2(n) W_{N/2}^{mk} \quad (2.21)$$

$$X(k+N/2) = X_1(k) - W_N^k X_2(k), k = (0,1,2,.....N/2) \quad (2.22)$$

Hence, the total number of complex multiplications for X(k) is 2·(N/2)2 +N/2=N2/2+N/2. For original N-point DFT equation (2.14), it has N2 complex multiplications.Then in the first step, separating x(n) into two parts makes the number of complex multiplications from N2 to N2/2+N/2. The number of calculations is reduced by approximately half

This is the process to reduce the calculations from N points to N/2 points. So continuously separating the x1(m) and x2(m) independently into the odd partand the even part in the same way, the calculations for N/2 points will be reduced for N/4 points.So,the calculations of DFT will be

continuously reduced.If the signal for N-point DFT is continuously separated until the final signal sequence is reduced to the one point sequence.

Let,there are $N = 2^s$ points DFT needed to be calculated.Then the number of such separations can be done is $s = log_2(N)$. So the total number of complex multiplications will be approximately reduced to $(N/2)log_2(N)$. For the addition calculations, the number will be reduced to $Nlog_2(N)$ As the multiplications and additions are reduced, the speed of the DFT computation is improved. The basic idea for Radix-2 FFT is to separate the old data sequence into odd part and even part continuously to reduce approximately half of the original calculations.

2.4 Frequency analysis in MATLAB of Speech Recognition:

Spectrum Normalization:

After DFT and FFT calculations are completed, the investigated problems will be changed from the discrete-time signals x(n) to the frequency domain signal $X(\omega)$

The spectrum of the $X(\omega)$ is the total integral or the summation of the all frequency components. When We talk about the speech signal frequency for different speech words, each speech-word has its frequency band, not just a single frequency.In the frequency band of each word, the spectrum $(|X(\omega)|)$ or spectrum power $(|X(\omega)|^2)$ has maximum value and minimum value. When comparing the differences between two different speech signals, it is hard or unconvincing to compare two spectrums in different measurement standards. So using the normalization can make the measurement standard the same.

Generally, the normalization can reduce the error when comparing the spectrums, that is good for the speech-recognition . So before analyzing the spectrum differences for different words, the first step is to normalize the spectrum $(|X(\omega)|)$ by the linear normalization. The equation for the linear normalization is given below:

$$y = (x - MinValue)/(MaxValue - MinValue) \qquad (2.23)$$

After normalization, the values of the spectrum $(|X(\omega)|)$ set into interval [0, 1]. The normalization only changes the values' range of the spectrum, but never changes the shape or the information of the spectrum itself. So the normalization is very good for spectrum comparison.Using MATLAB gives an example to see how the spectrum is changed by the linear normalization. Firstly, record a speech signal and do the FFT of the speech signal.And Then take the absolute values of the FFT spectrum.

2.5 The Cross-correlation Algorithm:

There is a substantial amount of data on the frequency of the voice fundamental (F_0) in the speech of speakers who differ in Sex and age. For the same speaker, the different words also have the different frequency bands which are due to the different vibrations of the vocal cord. Also the shapes of spectrums are also different.

These are the bases of this thesis for the speech recognition. In this thesis, to realize the speech recognition, We compare spectrums between the third recorded signal and the first two recorded reference signals. By checking which of two recorded reference signals better matches the third recorded signal,Then the system will give the judgment that which reference word is once again recorded at the third time. When We think about the correlation of two signals, the first algorithm that will be considered is the cross-correlation of two signals. The cross-correlation function method is really useful to estimate shift parameter. Here the shift parameter will be referred as- frequency shift

$$r_{xy} = r(m) = \sum_{n=-\infty}^{\infty} x(n)\,y(n+m), (m = 0,1,2,.....) or (m = 0,-1,-2.....) \qquad (2.24)$$

Mathematically, if we Define the frequency spectrum's function as a function f(x), according to axial symmetry property definition: for the function f(x), if x1 and x3 are axis-symmetric about x=x2, then f(x1) =f(x3). In the speech recognition comparison, after calculating the cross-correlation of two recorded frequency spectrums, there is obviously need to find the position of the maximum value of the cross-correlation.We use the values right to the maximum value position to minus the values left to the maximum value position. Taking the absolute value of this difference and find the mean square-error of the absolute value. When the two signals matched better, the cross-correlation is more symmetric. Also if the cross-correlation is more symmetric, then the mean square-error would be smaller. By comparing of this error, the system take finest decision that which reference word has been recorded at the third time.

The two important information of the cross-correlation may be given.First One is when two original signals have no time shift,then their cross-correlation would be the maximum; the Second information is that the position difference between the maximum value position and the middle point position of the cross-correlation is the length of time shift of two original signals.

Now assume that the two recorded speech signals for the same word are actually the same, then the spectrums of two recorded speech signals are also the same.So whenever doing the cross-correlation for the two same spectrums and plotting the cross-correlation, the graph of the cross-correlation should be symmetric according to the cross-correlation Algorithm.In case of actual speech recording, the spectrums of twice recorded signals that are recorded for the same word cannot be the totally the same. But their spectrums should be similar, that means their cross-correlation graph would be approximately symmetric. It is the most important and significant idea in my thesis for the speech recognition when I design the system 1.

By comparing the level of symmetric property of the cross-correlation, the system can take the decision that which two recorded signals have more similar spectrums. In other words, these two recorded signals are possibly recorded for the same word.

First two recorded reference speech are "hahaha" and "meat".The third time recorded speech word is "hahaha" once again. From Fig.10, the first plotting is the cross-correlation between the third recorded speech signal and the reference signal "hahaha". The second one plotting is the cross-correlation between the third recorded speech signal and the reference signal "meat". Since the third speech word is "hahaha",The first plotting is actually more symmetric as well as smoother than the second plotting.

2.6 The Auto-Correlation Algorithm:

In previous part, it is about the Algorithm of cross-correlation. Look at the the equation (2.24), the autocorrelation may be treated as computing the cross-correlation of the signal and itself instead of two different signals. This is the definition of auto-correlation in MATLAB. The auto-correlation is the algorithm to measure how the signal is self-correlated with itself.

The equation of Auto-Correlation is:

$$r_x(k) = r_{xx}(k) = \sum_{k=-\infty}^{\infty} x(n)x(n+k) \tag{2.25}$$

2.7 The FIR Wiener Filter:

FIR Wiener filter is used to estimate the desired signal d (n) from the observation process x (n) to get the estimated signal d (n)'.We assume that d (n) and x (n) are correlated and jointly wide-sense stationary. And the error of estimation is e (n) =d (n)-d (n)'.

So the output d (n)' is the convolution of x (n) and w (n):

$$d(n)' = w(n) * x(n) = \sum_{l=0}^{p-1} w(l)x(n-1) \tag{2.26}$$

then the error estimation is:

$$e(n) = d(n) - d(n)' = d(n) - \sum_{l=0}^{p-1} w(l)x(n-l) \tag{2.27}$$

The main purpose of FIR Wiener filter is to choose the suitable filter order and then find the filter coefficients with which the system can get the best estimation.Hence,with the proper coefficients the system can minimize the mean-square error:

$$\xi = E\,|e(n)|^2 = E\,|d(n) - d(n)'|^2 \tag{2.28}$$

Minimize the mean-square error to get the suitable filter coefficients, there is a sufficient method for doing this is to get the derivative of ξ to be zero with respect to $w^*(k)$. As the following equation:

$$\delta\xi(\xi * k) = \delta/(\delta w * k)Ee(n)e^*(n) = Ee(n)\delta e^*(n)/(\delta w^*(k) = 0 \tag{2.29}$$

From the equation (2.27) and (2.29) we know:

$$\delta e^*(n)/\delta w * k = -x^*(n-k) \tag{2.30}$$

$$\delta\xi\,\delta w^*(k) = Ee(n)\delta e^*(n)/\delta w^*(k) = -Ee(n)x^*(n-k) = 0 \tag{2.31}$$

$$Ee(n)x^*(n-k) = 0, x = (0,1,2,...., p-1) \tag{2.32}$$

$$Ee(n)x^*(n-k) = E[d(n) - \sum_{l=0}^{P-1} w(l)x(n-l)]x^*(n-k) = 0 \tag{2.33}$$

$$Ed(n)x^*(n-k) - E\sum_{l=0}^{p-1} w(l)x(n-1)x^*(n-k) = r_{dx} - \sum_{l=0}^{p-1} w(l)r_x(k-1 = 0 \tag{2.34}$$

Finally the equation becomes:

$$\sum_{l=0}^{p-1} w(l)r_x(k-1) = r_{dx} \tag{2.35}$$

with

$$r_x(k) = r_x*(-k)$$

,the Above equation may be in matrix form:

$$\begin{bmatrix} r_x(0) & r_x*(1) & ... & r_x*(p-1) \\ r_x(1) & r_x(0) & ... & r_x*(p-2) \\ r_x(2) & r_x(1) & ... & r_x*(p-3) \\ ..... & ..... & ... & ..... \\ ..... & ..... & ... & ..... \\ r_x(p-1) & r_x(p-1) & ... & r_x(0) \end{bmatrix} \begin{bmatrix} w(0) \\ w(1) \\ w(2) \\ .... \\ .... \\ w(p-1) \end{bmatrix} = \begin{bmatrix} r_{dx}(0) \\ r_{dx}(1) \\ r_{dx}(2) \\ ..... \\ ..... \\ r_{dx}(p-1) \end{bmatrix} \quad (2.36)$$

The matrix equation (2.36) is actually Wiener-Hopf equation of:

$$R_x w = r_{dx} \quad (2.37)$$

In this thesis, the Wiener-Hopf equation can work well for the voice recognition. From equation (37), the input signal x (n) and the desired signal d (n) are only the things that need to know. Then using x (n) and d(n) finds the cross-correlation r_{dx}.

simultanously, using x (n) finds the auto-correlation $r_x(n)$ and using $r_x(n)$ forms the matrix R_x in MATLAB. When having the $R_x and r_{dx}$ it can be directly found out the filter coefficients. With the filter coefficients it can get the minimum mean square-error ξ continuously from equations (27), (28), and (32), the minimum mean square-error ξ is:

$$\xi_{min} = Ee(n)d*(n) = E[d(n) - \sum_{l=0}^{p-1} w(l)x(n-l)]d*n = r_d(0) - \sum_{l=0}^{p-1} w(l)r_{dx}*(l) \quad (2.38)$$

We have to Apply the theory of Wiener filter to the speech recognition. If you want to use the Wiener-Hopf equation, it is mandatory to know two given conditions: First one is the desired signal d (n); the next one is the input signal x(n).In my thesis, it is to be assumed that the recorded signals are wide-sense stationary processes.So the first two recorded reference signals are used as the input signals x1(n) and x2(n).And the third recorded speech signal used as the desired signal d(n).This is a wish to find the best estimation of the desired signal in the Wiener filter.So the procedure to apply Wiener filter to the speech recognition can be thought as using the first two recorded reference signals to estimate the third recorded desired signal.hence, one of two reference signals x1(n), x2(n) is recorded for the same word as the word that is recorded at the third time.However, using the one of two reference signals that is recorded for the same word as the third time recording to be the input signal of Wiener filter will have the smaller estimation minimum mean square-error ξ_{min} according to equation (2.38).

However,After defining the roles of three recorded signals in the designed system 2, the next step is just to find the auto-correlations of the reference signals, which are $r_{x1}(n)$ and $r_{x2}(n)$ and find the cross-correlations for the third recorded voice signal with first two recorded reference signals, that are $r_{dx1}(n) and r_{dx2}(n)$. And use $r_{x1}(n), r_{x2}(n)$ to build the matrix $R_{x1} and Rx2$. lastly, according to the Wiener-Hopf equation (37),We have to calculate the filter coefficients for both two reference signals and also find the mean values of the minimum mean square-errors with respect to the two filter coefficients.After Comparing the minimum mean square-errors, the system will give the judgment that which one of the two recorded reference signals will be the word that is

recorded at the third time.So, The better estimation, the smaller mean value of ξ_{min}.

2.8 Use spectrogram Function in MATLAB to Get Desired Signals:

The spectrogram is a time-frequency plotting that contains power density distribution at the same time with respect to both of frequency axis and time axis. In MATLAB, this is easy to get the spectrogram of the voice signal by defining some variables: the sampling frequency, the length of window and the length of Shot-Time Fourier Transform (STFT). In the previous parts of this paper, FFT and DFT have been introduced. The STFT is firstly to use the window function to truncate the signal in the time domain, that makes the time-axis into several parts. If the window is a vector, then the number of parts is equal to the length of the window. Then compute the Fourier Transform of the truncated sequence with defined FFT length (nfft).

The Fig. 5.17 the spectrogram for the recorded speech signal in MATLAB, with the defined fs=16000, nfft=1024, the length of hanning window is 512, and the length of overlap is 380.So, It is necessary to mention that the length of window has to be smaller or equal than 1/2 the length of the STFT (nfft) when programming in MATLAB.

ISSN No:-2456-2165

CHAPTER-3
REVIEW

Programming Steps

3.1 Programming Steps for Designed System-1:

(1) Firstly, Initialize the variables and set the sampling frequency as:fs=16000. Also,Use "wavrecord" command to record Three voice signals. Make the first two recordings as the reference signals.And Make the third voice recording as the target signal.

(2) Write the function-"spectrogram" to process recorded signals and get returned matrix signals.

(3) Then,Transpose the matrix signals for rows and columns, take "sum" operation of the matrix and get a returned row vector for every column summation result. This row vector is the frequency spectrum signal.

(4) Normalize the frequency spectrums by linear normalization.

(5) Do the cross-correlations for the third recorded signal with the first two recorded reference signals separately.

(6) This step is very important as the comparison algorithm is programed here. Firstly, check the frequency shift of the cross-correlations. Now, it must be announced that the frequency shift is not the real frequency shift. This is the processed frequency in MATLAB. By the definition of the spectrum for the "nfft", which is the length of the STFT programmed in MATLAB, the function must return a frequency range that is respect to the "nfft". If "nfft" is odd, so the returned matrix has $(nfft+1)/2$ rows.if “nfft” is even, then the returned matrix has $(nfft/2)+1$ These are defined in MATLAB. Rows of the returned "spectrogram" matrix are still the frequency ranges. If the difference between the absolute values of frequency shifts for the two cross-correlations is greater or equal than 2, then the system will give the judgment only by the frequency shift.hence, The smaller frequency shift means the better match. If the difference between the absolute values of frequency shifts is smaller than 2, then the frequency shift difference is useless according to the experience by large amounts tests.And The system needs continuously do the comparison by the symmetric property for the cross-correlations of the matched signals. According to the symmetric property, MATLAB will give the judgment.

3.2 Programming Steps for Designed System-2:

(1)Firstly, Initialize the variables and set the sampling frequency fs=16000.

(2) Use "wavrecord" to record 3 voice signals. Make the first two recordings as the reference signals. Make the third voice recording as the target voice signal.

(3) Use the function "spectrogram" to process recorded signals and get the returned matrix signals.

(4)Now, Transpose the matrix signals for rows and columns, and take "sum" operation of the matrix and get a returned row vector for every column summation result. The row vector represents the frequency spectrum.

(5)The frequency spectrums must be normalized by the linear normalization.

(6) From this step, the system will be different to system 1. The system is to be programmed for the winner filter mode. Firstly calculate the auto-correlations of three signals: the first two recorded reference signals and the third recorded target signal. Second, set the total order number is 20. And use a “for” loop to detect each order result. For certain filter order, define the auto-correlation length from N to N+p. By the definition of the Wiener filter equation (2.36), the lengths of the auto-correlation matrix and auto-correlation vector both should be p. Since the position N is the maximum value position of the auto-correlation, so "r(N)=r(0)". To be more clearly, this is explained the part chapter-2, that introduced the relation between the maximum value and the position for the cross-correlation. After defining $R_x and r_{dx}$, the next step is to calculate directly the filter coefficients for each reference signal.

ISSN No:-2456-2165

(7) After We find the filter coefficients for each reference signal, We have to calculate the minimum mean square-error for each reference signal.Now, Compare the mean value of the minimum mean square-errors for the order range from 0 to 20. The better estimation must have the smaller minimum mean square-errors. The theory of the Wiener filter has been already introduced in the part of chapter-2.

ISSN No:-2456-2165

CHAPTER-4
MATERIALS & METHODS

4.1 Table 1

Test times	Frequency-on-shift	Frequency-off-shift	Error1	Error2	**Final judge-ments**
1	2	8	no need	no need	on
2	7	8	0.2055	0.4324	on
3	8	9	0.2578	0.2573	off
4	9	17	no need	no need	on
5	8	9	0.2304	0.3640	on
6	0	0	0.3268	0.6311	on
7	0	0	0.3193	0.3210	on
8	0	0	2.2153	0.9354	off
9	0	0	0.4603	0.1481	off
10	0	0	0.1189	0.0741	off
11	8	22	no need	no need	Door
12	8	0	no need	no need	key
13	8	25	no need	no need	Door
14	8	24	no need	no need	Door
15	8	24	no need	no need	Door
16	-15	0	no need	no need	key
17	-15	0	no need	no need	key
18	-14	0	no need	no need	key
19	-14	0	no need	no need	key
20	-14	0	no need	no need	key

Total Successful Probability(Total in 20 times) =80 %

Table 4.1: Simulation results for speech words "On", "Off", "Door" and "Key"

ISSN No:-2456-2165

4.2 Table 2

Test times	Frequency-door-shift	Frequency-key-shift	Error1	Error2	Final judge-ments
1	-2	15	no need	no need	door
2	-1	15	no need	no need	door
3	0	14	no need	no need	door
4	0	14	no need	no need	door
5	0	8	no need	no need	door
6	-1	13	no need	no need	door
7	0	13	no need	no need	door
8	0	14	no need	no need	Door
9	0	13	no need	no need	door
10	0	15	no need	no need	door
11	-23	0	no need	no need	key
12	-8	0	no need	no need	key
13	-8	-1	no need	no need	key
14	-16	0	no need	no need	key
15	-16	0	no need	no need	key
16	-14	0	no need	no need	key
17	-20	0	no need	no need	key
18	-13	0	no need	no need	key
19	-13	0	no need	no need	key
20	-12	0	no need	no need	key

Total Successful Probability(Total in 20 times for each of "door" and "key") =100 %

Table 4.2: indicates the simulation results for reference signals “Door” and “Key” as the information given at the beginning of the Simulation Results from fig. 5.5 and fig 5.6

ISSN No:-2456-2165

4.3 Table 3

Test times	Frequency-on-shift	Frequency-off-shift	Error1	Error2	Final judge-ments
1	0	0	0.0888	0.2858	on
2	0	0	0.0979	0.2645	on
3	0	0	0.1073	0.3327	on
4	0	0	0.0430	0.1958	on
5	0	0	0.0075	0.0476	on
6	0	0	0.0885	0.1834	on
7	0	0	0.1121	0.0390	off
8	0	0	0.0281	0.1699	On
9	0	0	0.0755	0.0286	off
10	0	0	0.0389	0.3312	on
11	0	0	0.2289	0.0075	off
12	0	0	0.2316	0.1499	off
13	0	0	0.1519	0.0228	off
14	0	0	0.1123	0.0072	off
15	0	0	0.0240	0.0360	on
16	-1	0	0.2900	0.0245	off
17	0	0	0.1984	0.0162	off
18	0	-1	0.2414	0.526	off
19	-1	0	0.4284	0.0246	off
20	0	0	0.1334	0.0269	off

Total Successful Probability(Total in 20 times for "on") =80 %
Total Successful Probability(Total in 20 times for "off_") =90 %

Table 4.3: indicates the simulation results for reference signals “on” and “off” as the information given at the beginning of the Simulation Results fig. 5.8

ISSN No:-2456-2165

4.4 Table 4

Test times	Frequency-Door-shift	Frequency-Key-shift	Error1	Error2	Final judge-ments
1	-2	15	no need	no need	door
2	-1	15	no need	no need	door
3	0	14	no need	no need	door
4	0	14	no need	no need	door
5	0	8	no need	no need	door
6	-1	13	no need	no need	door
7	0	13	no need	no need	door
8	0	14	no need	no need	Door
9	0	13	no need	no need	door
10	0	15	no need	no need	door
11	-23	0	no need	no need	key
12	-8	0	no need	no need	key
13	-8	-1	no need	no need	key
14	-16	0	no need	no need	key
15	-16	0	no need	no need	key
16	-14	0	no need	no need	key
17	-20	0	no need	no need	key
18	-13	0	no need	no need	key
19	-13	0	no need	no need	key
20	-12	0	no need	no need	key

Total Successful Probability(Total in 20 times for "Door") =100%
Total Successful Probability(Total in 20 times for "Key") =100 %

Table 4.4:indicates the simulation results for reference signals "Door" and "key(noisy)" as the information given at the beginning of the Simulation Results fig. 5.9

In the Next Page-tables, "m1" is mean value of the minimum mean square-errorsfor reference signal-1. "m2" is the mean value of the minimum mean square-errors for reference signal-2. The definition of the mean value of the minimum mean square-errors described in the part FIR wiener filter(chapter 2) of this thesis.

ISSN No:-2456-2165

4.5 Table 5

Test times	m1	m2	Final- judge ments
1	1.6740	3.4707	on
2	1.4442	2.6448	on
3	1.9087	3.7704	on
4	1.5448	2.9563	on
5	1.6103	3.4758	on
6	1.3971	3.7114	on
7	1.9205	3.3964	on
8	1.4944	3.7154	on
9	1.4716	3.7707	on
10	1.7879	4.5664	on
11	3.4775	0.1948	off
12	2.9213	1.0938	off
13	2.3013	0.7820	off
14	2.8370	1.1304	off
15	2.2277	0.7933	off
16	2.8922	1.8087	off
17	3.7633	1.6842	off
18	3.3211	1.5603	off
19	2.5852	0.8402	off
20	3.2708	1.4720	off

Total Successful Probability(Total in 20 times for "on") =100 %
Total Successful Probability(Total in 20 times for "off") =100 %

Table 4.5: indicates the simulation results for reference signals "on" and "off"

ISSN No:-2456-2165

4.6 Table 6

Test times	m1	m2	Final judge-ments
1	0.1837	5.2746	Door
2	0.7070	6.5936	Door
3	0.7565	8.4193	Door
4	0.2680	5.5085	Door
5	0.2973	6.0271	Door
6	0.8471	7.5534	Door
7	0.7039	7.8490	Door
8	0.5523	7.5952	Door
9	0.2467	5.7104	Door
10	0.3792	4.8257	Door
11	6.7231	0.0990	Key
12	8.7500	1.0829	Key
13	6.0756	0.1670	Key
14	8.1771	0.5392	Key
15	7.0094	0.2012	Key
16	7.9720	0.7285	Key
17	6.4771	0.4326	Key
18	6.3291	0.6853	Key
19	5.2181	0.4563	Key
20	5.1493	0.1231	off

Total Successful Probability(Total in 20 times for "Door") =100 %
Total Successful Probability(Total in 20 times for "key") =100 %

Table 4.6: indicates the simulation results for reference signals "Door" and "Key"

ISSN No:-2456-2165

Chapter-5
Results & Discussion

5.1 Simulation Results:

Here, I simulated two designed systems within the help of my friends who are from different Places or different countries.Since the thesis introduced previously, the only task of operator is to run the program and record three speech or voice signals. The first two recordings are used as reference signals.And The third recording used as the target signal for which MATALB would give the judgment. In the following results,I uses "reference signals" to stand for the first two recordings and uses "target signal" to stand for the third recording. The words in the quotes stand for the contents of recordings.I tried to test designed systems for both easily recognized words and difficulty recognized words. "From time 1 to 10, "on" in the following of the thesis means the operator simulated 10 times and the third recording word is “on” in the first 10 times simulations. Both the contents of the reference words and the target word are known, I want to test if the judgment that is given by MATLAB is correct as we known. The statistical simulation results are put in tables and will be plotted. In this Simulation Results portion, only the plotted results will be shown in the following content. As I programmed in MATLAB to plot figures for each system to help the analysis when simulating at every time, and the resulting figures for each system are got by the same principles, so I will put the simulation figure once time at the beginning of simulation results for each of the two systems.

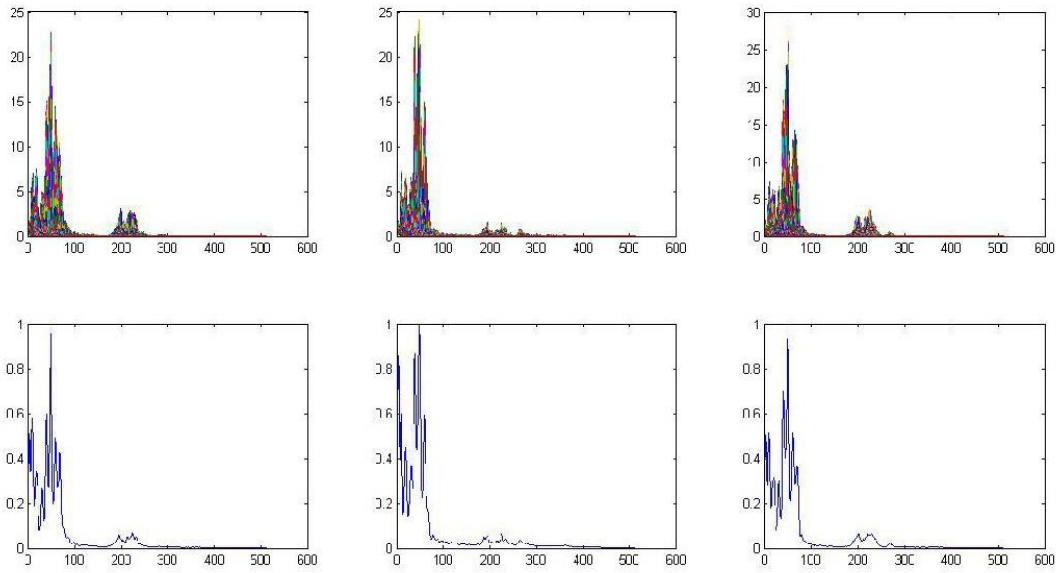

Fig. 5.1: Frequency spectrums for three speech signals: "on", "off", "on" for designed system-1

ISSN No:-2456-2165

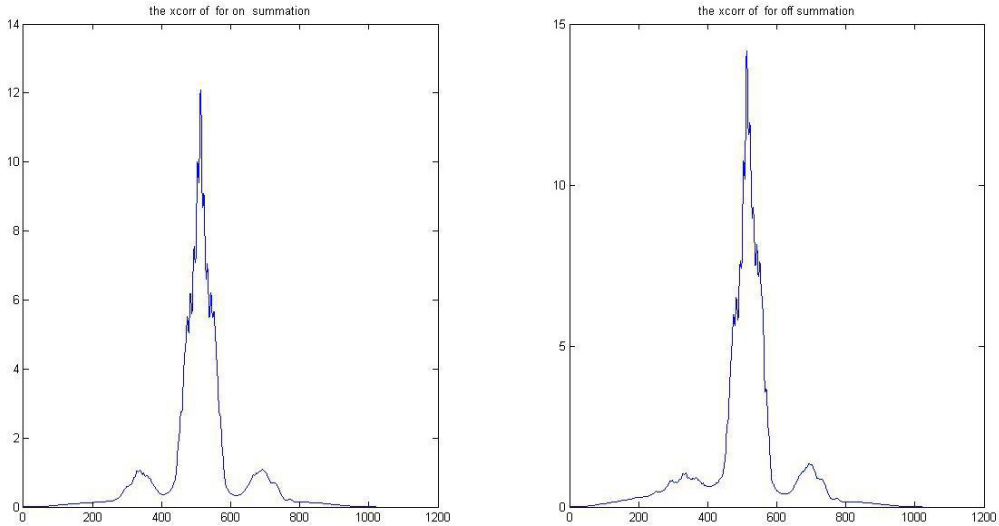

Fig. 5.2: Cross-correlations between the target signal "on" and reference signals for designed system-1

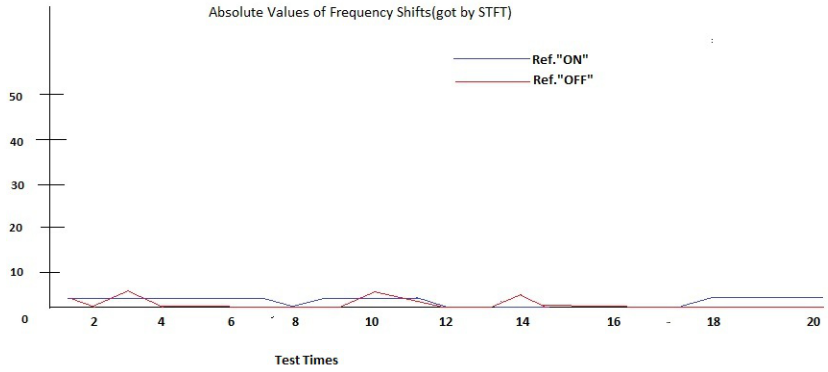

Fig. 5.3: Frequency shifts in 20 times simulations for reference "on" and "off" for designed system-1

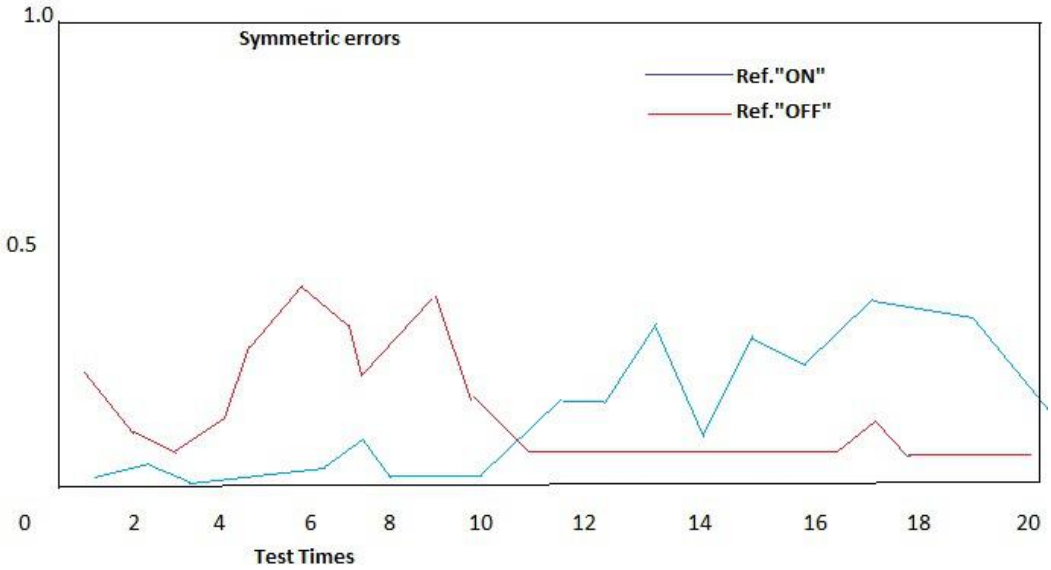

Fig. 5.4: Symmetric errors in 20 times simulations for reference "on" and "off" for designed system-1

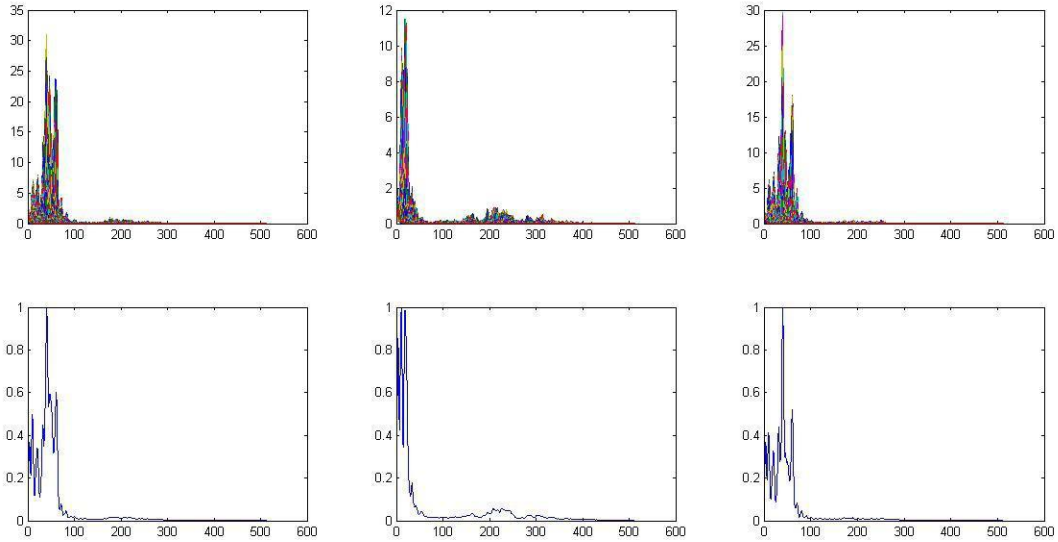

Fig. 5.5: Frequency spectrums for three signals: "Door", "Key", and "Door" for designed system-1

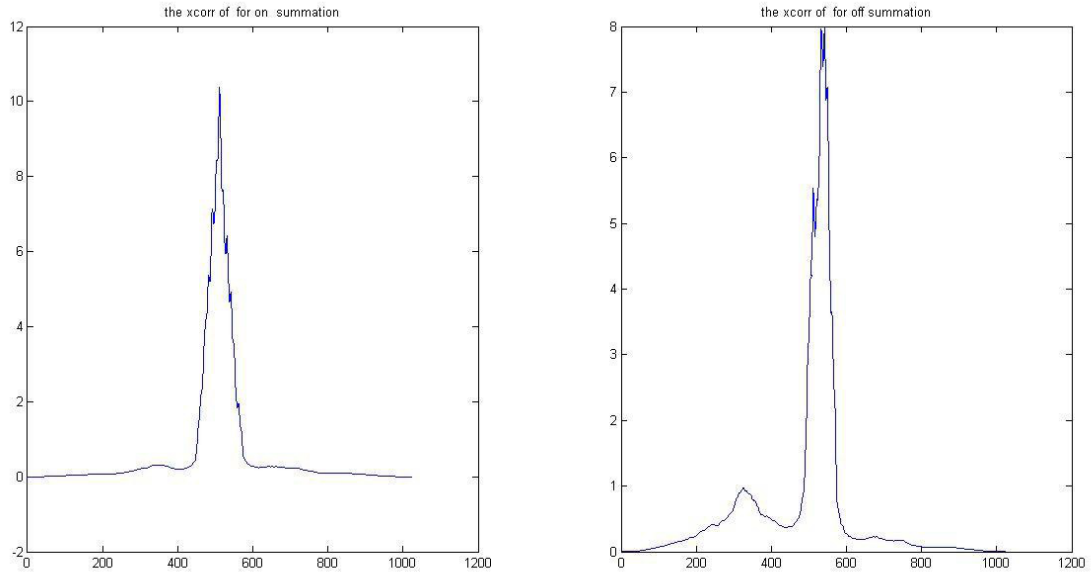

Fig. 5.6: Cross-correlations for the target signal "Door" with reference signals

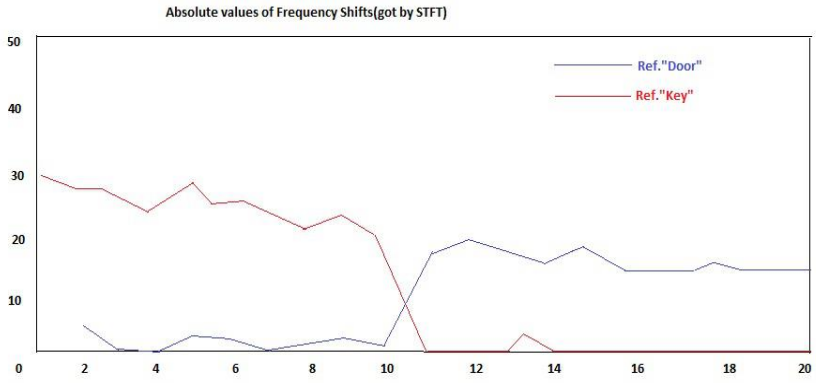

Fig. 5.7: Frequency shits in 20 times simulations for reference "Door" and "Key"for designed system-1

ISSN No:-2456-2165

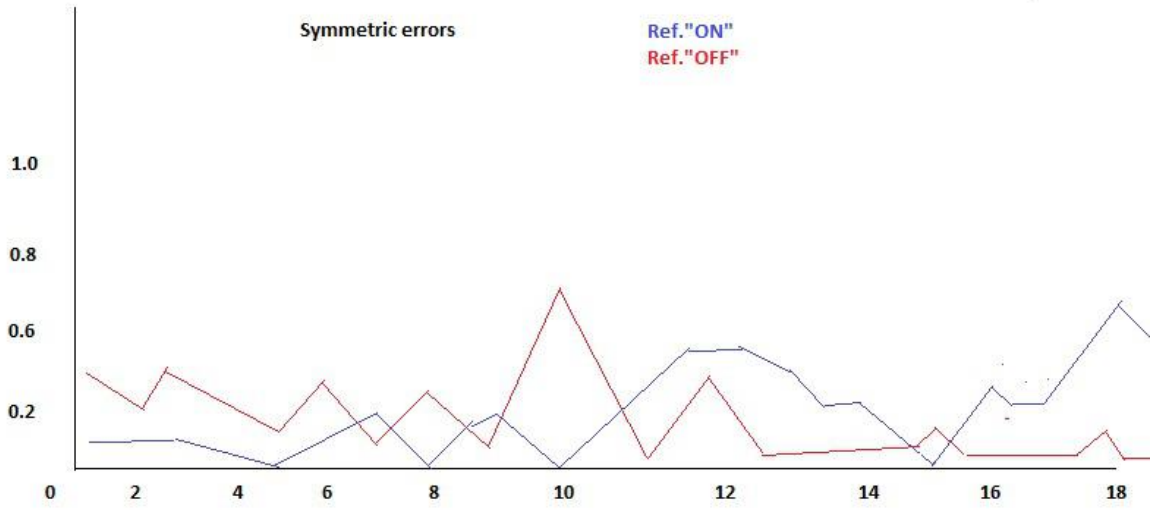

Fig. 5.8: Symmetric errors in 20 times simulations for reference "on" and "off" (noisy) for designed system-1

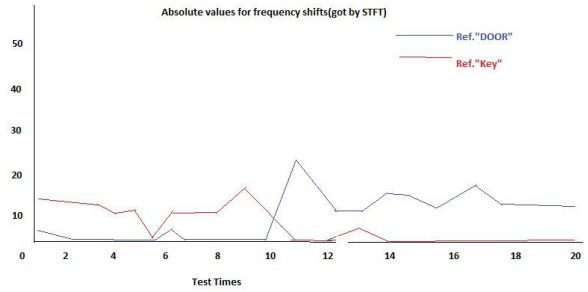

Fig. 5.9: Frequency shits in 20 times simulations for reference "Door" and "Key" (noisy) for designed system-1

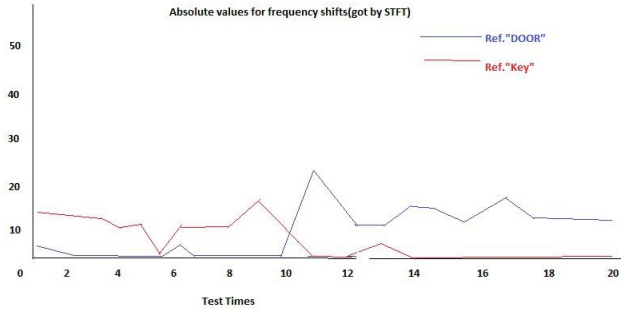

Fig. 5.10: Frequency shits in 20 times simulations for reference "Door" and "Key" (noisy) for designed system-1

ISSN No:-2456-2165

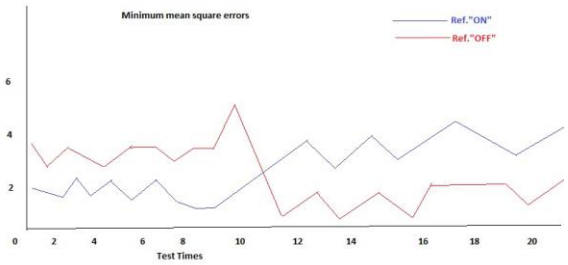

Fig. 5.11: Minimum mean square-errors for the target signals with reference signals-"on" and "off" for designed system-2

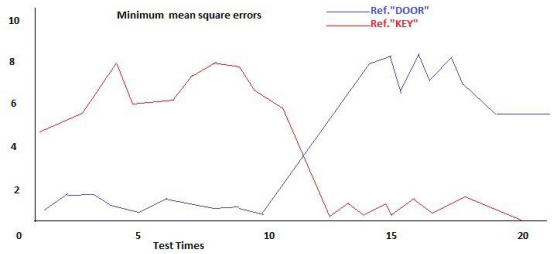

Fig. 5.12: Minimum mean square-errors for the target signals with reference signals-"Door" and "key" for designed system-2

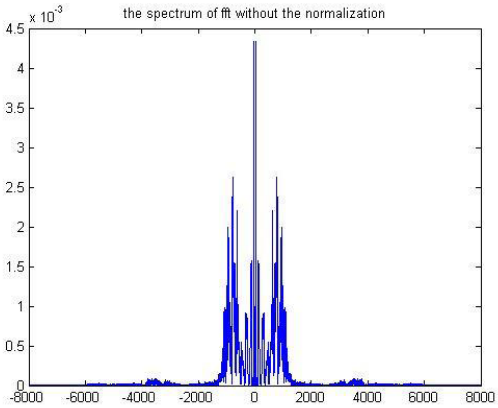

Fig. 5.13: Absolute values of the FFT spectrum without normalization

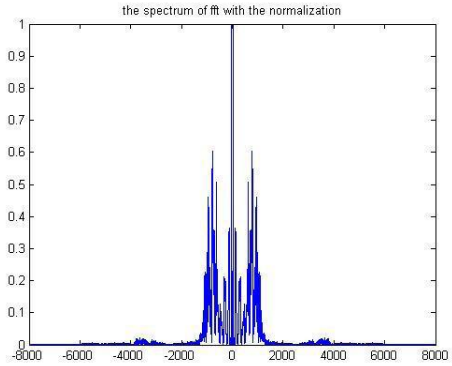

Fig. 5.14: Absolute values of the FFT spectrum with normalization

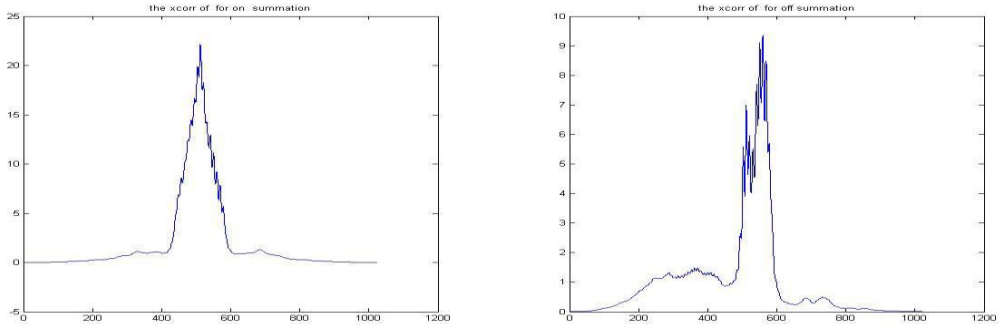

Fig. 5.15: The graph for Cross-Correlation

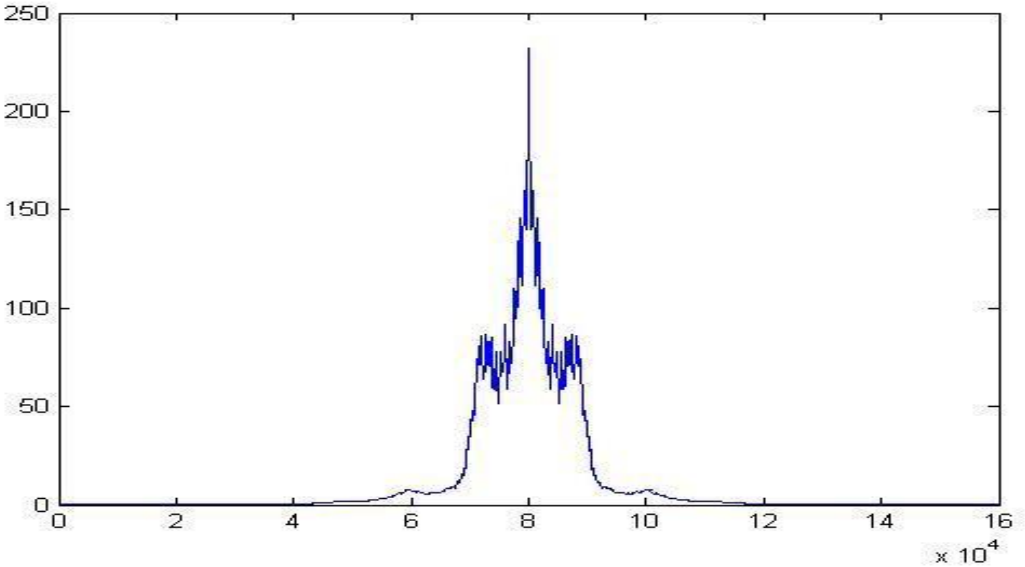

Figure 5.16: Auto Correlation for $|X(\omega)|$

ISSN No:-2456-2165

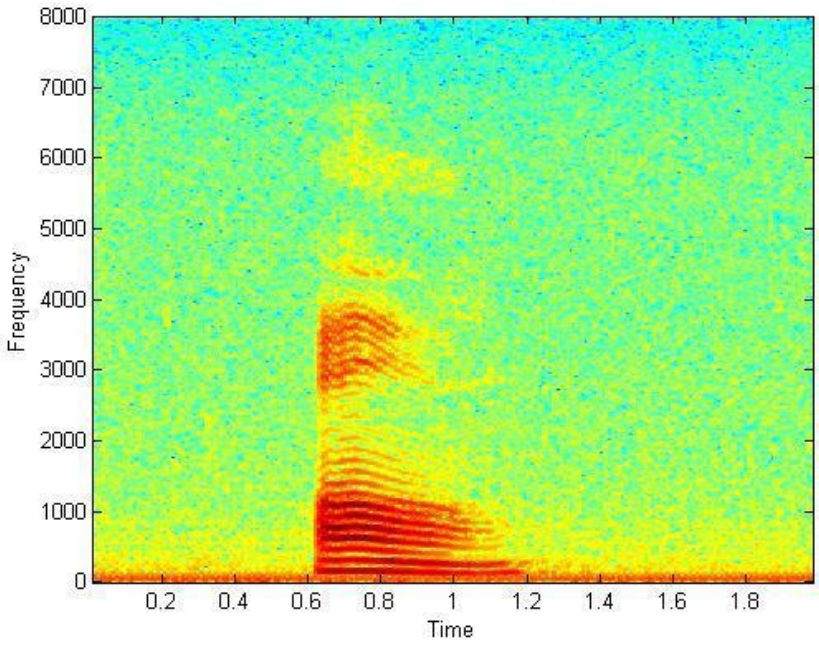

Fig. 5.17: The Spectrogram for recorded speech word "on"

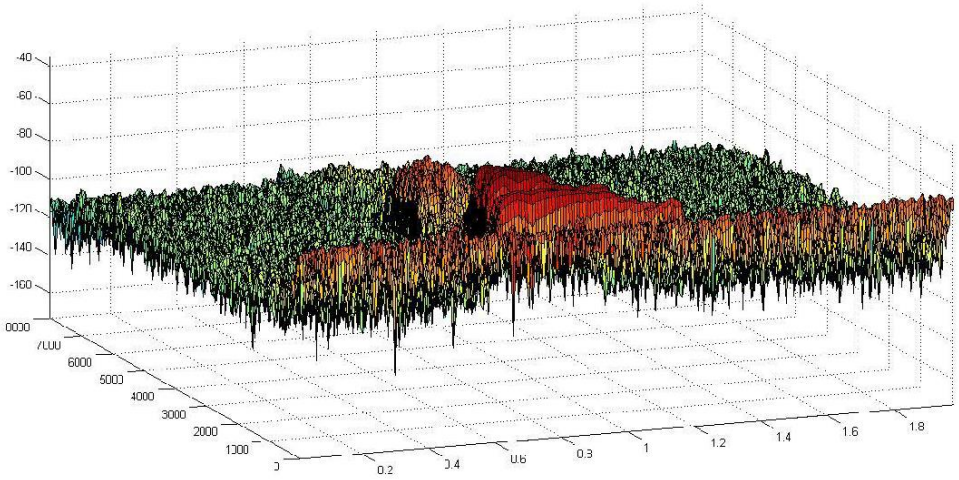

Fig. 5.18: The 3-Dimension Spectrogram for recorded speech word "on"

5.2 DISCUSSION:

- *Discussion about The Simulation Results for The designed System-1:*

There are 5 figures for the result figures in this chapter for the designed system 1. Table 1(fig 5.1) and Table 2(fig 5.2) are simulated by me in order to observe how the designed system works for the different pairs of words, the easily recognized words and also the difficultly recognized words. from the (Fig. 5.3), it can be observed that the frequency shift is not reliable, as the pronunciations of "on" and "off" are really closed. At this situation, the designed system must give the judgments by comparing the errors of the symmetric property of the cross-correlations (Fig. 5.4). If the pronunciations of two words are totally different, like "Door" and "Key", the designed system 1 make the judgments directly by the frequency shifts, that we can observe from (Fig. 5.7). The designed system does not calculate the errors anymore. By observing large amounts of simulation results, I programmed the system in MATLAB to rely frequency shifts when the difference between the absolute values of frequency shifts for the different reference signals is greater or equal to 2. Otherwise, the designed system-1 continuously calculate the errors of the symmetry. (Fig. 5.8) and (Fig. 5.9) are simulated by the my friends who are from different places. The purpose of these results is to show that the designed system actually works for the different people who are from different places. And from the(Fig. 5.8) and (Fig. 5.9), it can be seen that the designed system didn't work that well when there is noise around. This shows that the designed system can be disturbed easily by the noise. The figures 5 results are simulated by two people. One is responsible for the reference signals' recordings, the other one is for the target signal's recordings. It can be observed that the designed system almost doesn't work for this situation.

- *Discussion about The Simulation Results for The designed System-2*:

The main objectives of these simulations are the same as the designed purposes for the system1. The system-1 has been designed by observing large amounts of the plots of the cross-correlations. The system-2 is designed by using the reference signals to model the target signal. By comparing the errors between the real target signal and the modeling target signal got from the different reference signals, the system 2 gives the judgment that which reference signal is more similar to the target signal. I designed Wiener filter to realize this signal modeling process. In this system, the reference signals are being used as the auto-correlation sources, that are the inputs of the Wiener filter. And the target signal is being used as the desired signal. By the Wiener filter equation $R_x W = r_{dx}$, it can be known that when applying this equation, it actually gives the assumption that the input signal x (n) is correlated to the desired signal d (n). In other words, the reference signals should be correlated to the target signal. But if one person gives reference signals and the another person gives the target signal, then the reference signals and the target signal are not correlated to each other. So,it can be said that the designed system 2 doesn't work well when the reference signals and the target signal are recorded by different people.

ISSN No:-2456-2165

CHAPTER- 6
CONCLUSIONS

In general conclusions, the designed systems for speech recognition easily disturbed by noise, that can be observed from (Fig. 5.8). For the designed system 1, the better matched signals have better symmetric property of their cross-correlation. This conclusion can be got from (Fig. 5.3 and Fig. 5.4) and (Fig. 5.7).For the designed system 2, if the reference signal is the same word as the target signal, then using this reference signal to model the target will have less errors. This conclusion can be proved by the all the simulation results for the designed system 2. When both reference signals and the target signal are recorded by the same person, two systems work well for distinguishing different words, no matter where the person is from. But if the reference signals and the target signal are recorded by the different people, both systems(system-1 and system-2) don't work that well. So in order to improve the designed systems to make it work better, the further tasks are to enhance the systems' noise immunity and to find the common characteristics of the speech for the different people. Otherwise, designing some analog and digital filters for processing the input signals can reduce the effects of the input noise and to establish the large data base of the speech signals for different words.Studying more advanced algorithms for signal modeling can give a lot of helps to realize the better speech recognition.

ISSN No:-2456-2165

CHAPTER- 7
FUTURE WORKS

All speech recognition technology are very applicable to real life.But according to my thesis for three times of recording of voice should be the same person.otherwise the system will not work.That means out of three times of recording voice person should be the same...if different person's voice is present for each of the recording the system will reject the request. So, my designed system will be very applicable/useful or efficient for security purposes.

BIBLIOGRAPHY

[1] John G.Proakis, Dimitris G.Manolakis, Digital Signal Processing, Principles, Algorithms, and Applications, 4th edition,Pearson Education inc., Upper Saddle River.

[2] Joseph Picone, "Signal Modeling Techniques In Speech", Systems and Information Sciences Laboratory, Tsukuba Research and Development Center, Tsukuba,Japan.

[3] Luis Buera, Antonio Miguel, Eduardo Lleida, Oscar Saz, Alfonso Oretega, "Robust Speech Recognition with On-line Unsupervised Acoustic Feature Compensation", Communication Technologies Group (GTC),13A, University of Zaragoza, Spain.

[4] Hartmut Traunmüller , Anders Eriksson, "The frequency range of the voice fundamental in the speech of male and female adults", Institutionen för lingvistik, Stockholms universitet, S-106 91 Stockholm, Sweden.

[5] John Wiley, Sons,Inc. Statistical Signal Processing And Modeling, Monson H Hayes,Georigia I nstitute of Technology.

[6] Jian Chen, Jiwan Gupta,"Estimation of shift parameter of headway distributions using crosscorrelation function method", Department of Civil Engineering, The University of Toledo.

[7] Henrik V. Sorensen, C. Sidney Burrus, "Efficient Computation of the Short-Time Fast Fourier Transform", Electrical and Computer Engineering Department, Rice University, Houston.

[8] Fredric J. Harris,Member, IEEE, "On the Use of Windows for Harmonic Analysis with the Discrete Fourier transform", Proceedings of the IEEE, VOL 66, No.1, JANUARY, 1978

[9] Joseph W. Picone, senior member, IEEE "Signal Modeling Techniques in Speech" Recognition