

CHAPTER-1

INTRODUCTION

1.1 Introduction:

Speech has been one of the most potent tools at man's disposal since ancient times. Humans have constantly evolved and expressed themselves through speech. There is plethora of languages used and spoken by man throughout the entire world.

Computer speech recognition or Automatic Speech recognition system is a process in which the words spoken by humans or speech signals are translated into words. The words that are recognized by the machine can be the final output, it also uses algorithm, which is implemented as computer program. The major goal of Automatic Speech recognition is to characterize, extract and recognize these words as of some language. Speech technology focuses on development and improvisation of techniques that enables the computers to identify speech as an input effectively. Speech recognition technology has enormously evolved over the past few decades, we can now find it in phones, like features such as Siri, many of phone applications that are automated used by offices and airlines, and even applications that can be easily used at homes.

The speech signals, which are spoken, are actually quasi-stationary signals; hence we need to extract the features from speech. Feature extraction is the most integral phase of speech recognition. It is mainly used to reduce the noise and distortions in speech so that it can be effectively converted into text.

1.2 Speech recognition system:

Automatic speech recognition system enables a machine to hear, comprehend and respond to the speech signal, which is provided by the speaker as input. The methodology of the speech recognition system can be understood by the following flowchart i.e. Figure 1. The input is given to the system; this input is in the form of speech signal only wanted speech signals are retained. Then the feature extraction is performed and useful features are extracted out of the voice signal. After feature extraction next step is feature classification, the feature vectors are divided into two categories, training and testing. In training speech modeling is performed and in testing pattern matching is done. During recognition phase the test speech data is matched with the training models and the result is given according to the best match.

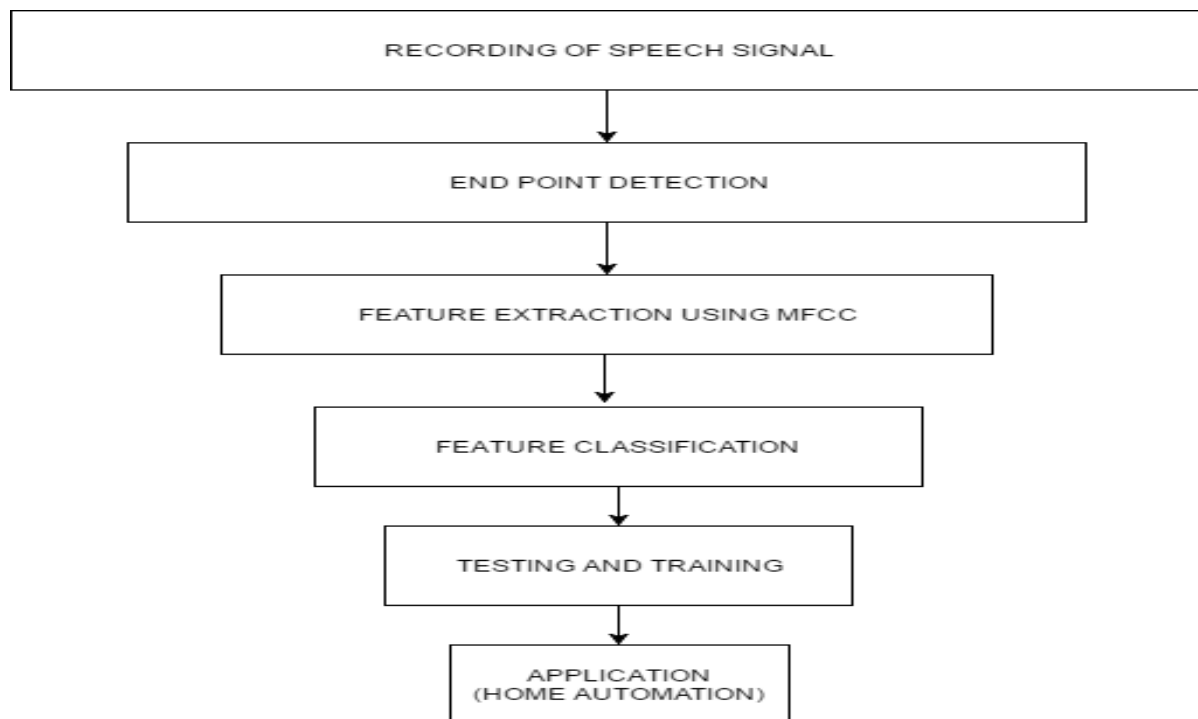


Fig 1.1 Project Methodology

1.3 Problem statement:

Speech recognition is a technology that able a computer to capture the words spoken by a human with a help of a microphone. These words are later on recognized by speech recognizer, and in the end, system outputs the recognized words. The process of speech recognition consists of different steps that will be discussed in the following sections one by one.

An ideal situation in the process of speech recognition is that, a speech recognition engine recognizes all the words uttered by a human but, practically the performance of a speech recognition engine depends on number of factors. Vocabularies, multiple users and noisy environment are the major factors that are counted in as the depending factors for a speech recognition engine.

1.4 Project Objectives:

The objective of the project is:

- 1.To develop an automatic speech recognizer.
- 2.To critically review literature related to Speech recognition. To identify speech corpus elements exhibited in English language.
- 3.To implement an isolated word speech recognizer that is capable of recognizing and responding to speech.
- 4.To develop real time speaker recognition system using feature extraction algorithms.
5. To design a real time home automation model.

1.5 Literature Survey:

1."An algorithm for determining the endpoints in isolated utterances", L.R.Rabiner and M.R.Sambur

Accurate location of the endpoints of an isolated word is important for reliable and robust word recognition. The endpoint detection problem is nontrivial for non-stationary backgrounds where artifacts (i.e., no speech events) may be introduced by the speaker, the recording environment, and the transmission system. Several techniques for the detection of the endpoints of isolated words recorded over a dialed-up telephone line were studied. The techniques were broadly classified as either explicit, implicit, or hybrid in concept. The explicit techniques for endpoint detection locate the endpoints prior to and independent of the recognition and decision stages of the system. For the implicit methods, the endpoints are determined solely by the recognition and decision stages of the system, i.e., there is no separate stage for endpoint detection. The hybrid techniques incorporate aspects from both the explicit and implicit methods. Investigations showed that the hybrid techniques consistently provided the best estimates for both of the word endpoints and, correspondingly, the highest recognition accuracy of the three classes studied. A hybrid endpoint detector is proposed which gives a rejection rate of less than 0.5 percent, while providing recognition accuracy close to that obtained from hand-edited endpoints.

2) "Speech Recognition using MFCC and DTW", Bhadragiri Jagan Mohan , Ramesh Babu N

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.

3) "Voice Controlled Automation System" Mohammad Salman Haleem.

In this era of technology, rapid advancements are being made in the field of automation and signal processing. The developments made in digital signal processing are being applied in the field of automation, communication systems and biomedical engineering. Controlling through human speech is one of the fascinating applications of digital signal processing (DSP) and Automation Systems. This paper discusses the speech recognition and its application in control mechanism. Speech recognition can be used to automate many tasks that usually require hands-on human interaction, such as recognizing simple spoken commands to perform something like

CHAPTER-2

END POINT DETECTION

2.1 Introduction:

Pre-Processing of Speech Signal is very crucial in the applications where silence or background noise is completely undesirable. Applications like Speech Recognition needs efficient feature extraction techniques from speech signal where most of the voiced part contains Speech specific attributes.

An important problem in speech processing is to detect the presence of speech in a background of noise. This problem is often referred to as the end point detection problem. By accurately detecting beginning and end of speech, the amount of processing speech data can be kept to minimum. The Task is to separating speech from background silence. In this Project we have used energy based end point detection algorithm, where first we divide the speech signal in to frames and finding energy of a frame & then retaining the frames which have energy which is greater than the threshold.

2.2 End point detection Algorithm:

Steps involved in the End point detection as follows:

STEP 1: Start the program and record the speech signal for the desired amount of time. $y(t)$

STEP 2: Sample the recorded signal. The sampling rate found to be 16000 Hz. $y(n)$

STEP 3: Sampled speech signal is divided into frames so, we have selected the frame size to be 800 samples and every time 400 samples

are overlapping with next frame. Each frame is multiplied with in-build matlab hamming window function.

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

Where $w(n)$ represents the hamming window.

$$X(i) = y(n) * w(n)$$

Where i represents the number of frames.

STEP 4: Find the energy of each frame,

$$e = \sum_{j=1}^L [X(i)]^2$$

Where L represents the frame size i.e. 800 samples for each frame.

STEP 5: Find the log energy of each frame and add 60dB

$$Le = 10 * \log_{10} e$$

$$Le = Le + 60dB$$

STEP 6: Here we are discarding the unwanted frames ,we have set the threshold to be around 40 dB so that if any frame which has the energy less than the 40 dB will be discarded and remaining frames are retained for post processing.

STEP 7: De-overlapping to get the end point detected signal.

Design of real time speech recognition system

FLOWCHART:

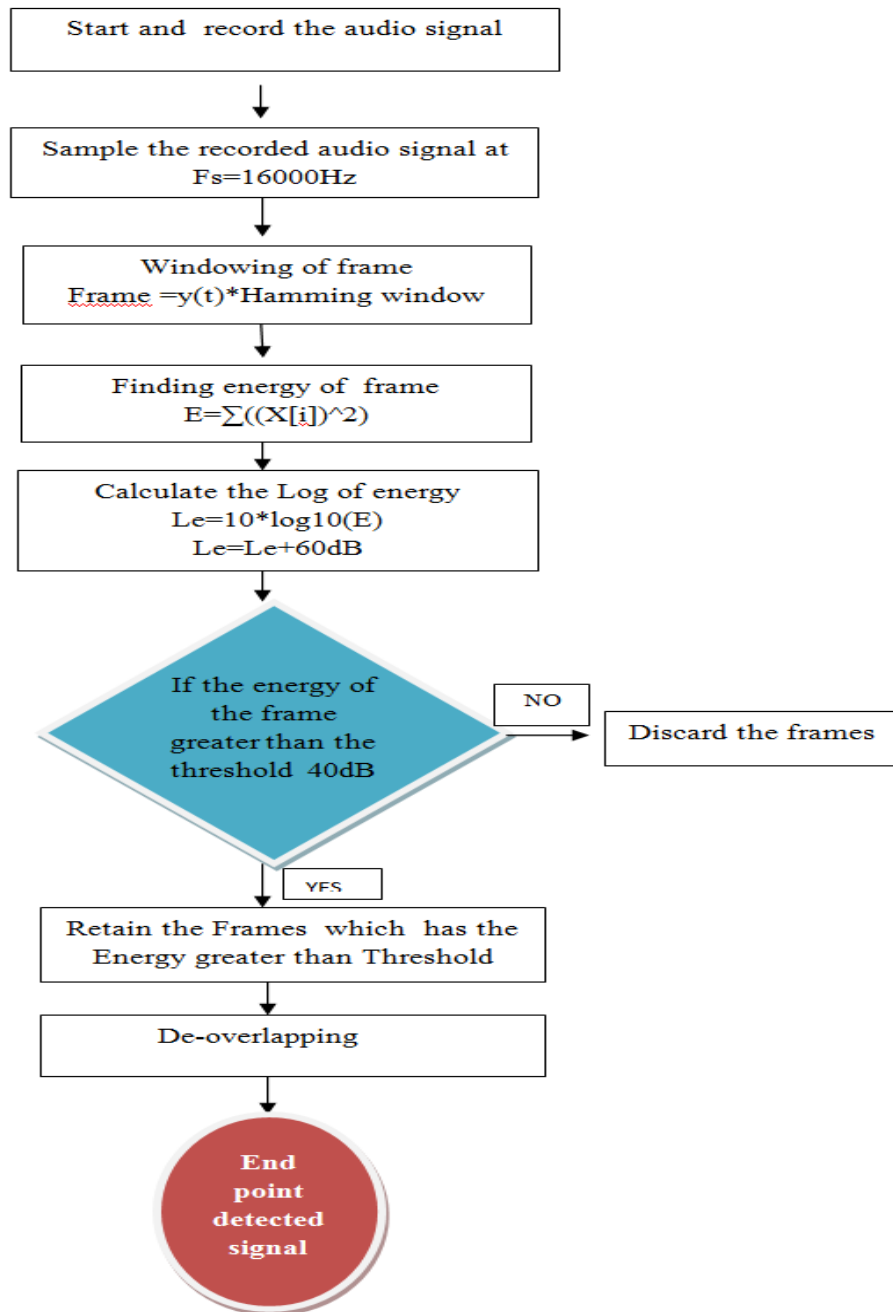


Figure 2.1 End point detection Flowchart

2.3 End point detection waveforms

Below shown figure is Amplitude Vs Time plot,
Figure 2.2 (a) is real time speech recorded signal for 8 seconds (the word uttered is 'GOOD MORNING').

Figure 2.2(b) represents the end point detected signal . We can observe that in time axis the amount of time is being reduced . So we can infer that we have avoided the unwanted processing samples.

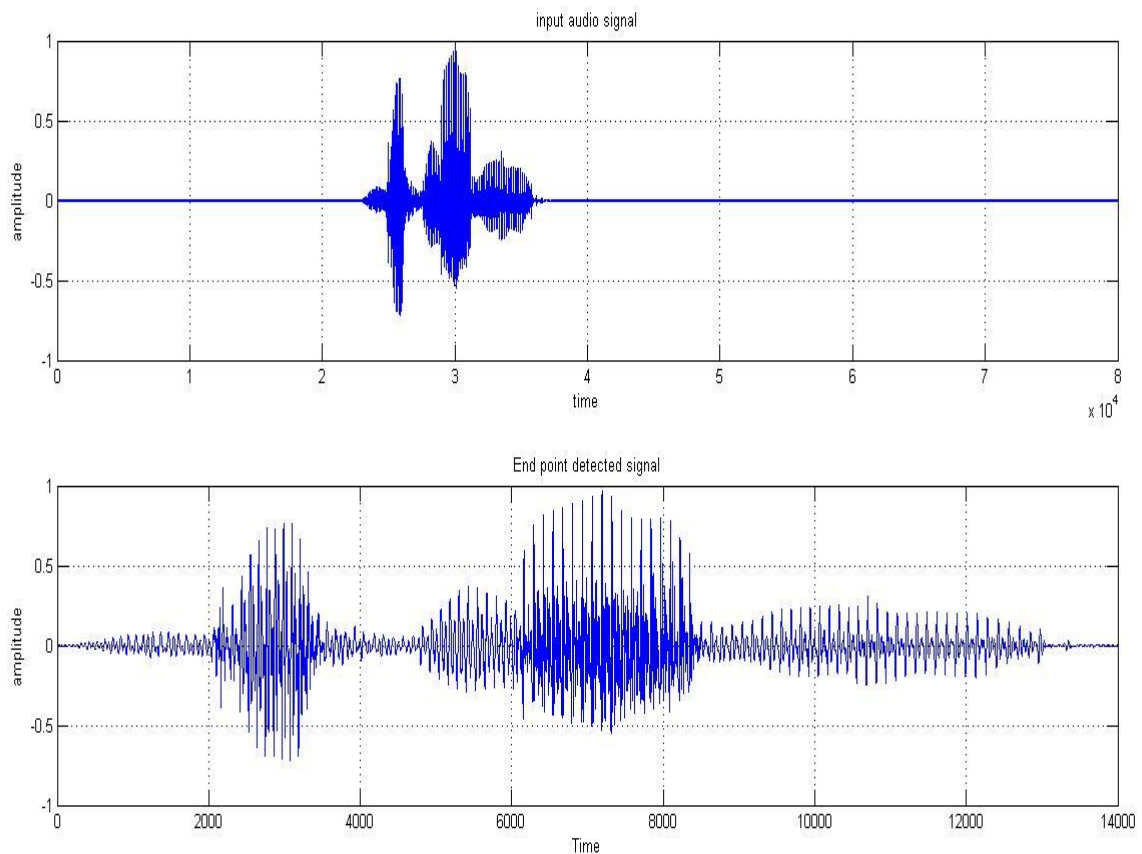


Fig 2.2 Input and output audio signals after end point detection

Figure 2.3(a) represents the energy plot of a input recorded signal for 8 seconds. Y-axis represents the log of energy in decibel scale, X-axis represents Number of frame. In each frame we have 800 samples.

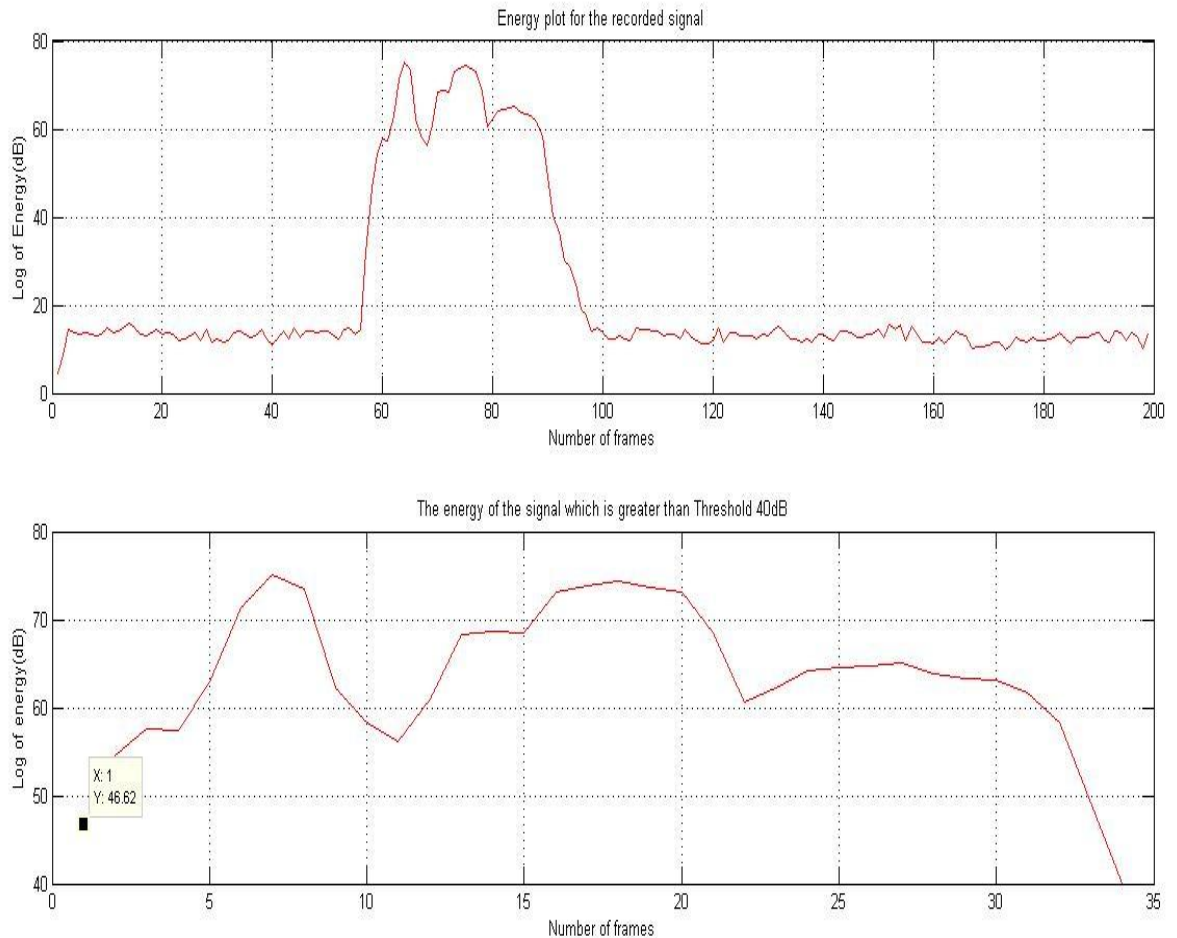


Figure 2.3 Energy plots of input and End point detected signal

Figure 2.3(b)

Represents the energy plot for frames which has energy greater than the threshold. In this figure we can see that the number of frames has reduced from 200 to 35. So from this the unwanted processing frames are reduced.

Figure 2.4 represents the complete waveforms involved in the process. Figure 2.4(a) shows the inputted real time audio signal, Figure 2.4 (b) shows the energy plot for the real time signal. Figure 2.4(c) shows the energy of the frames which has energy greater than threshold. Figure 2.4(d) represents the end point detected signal.

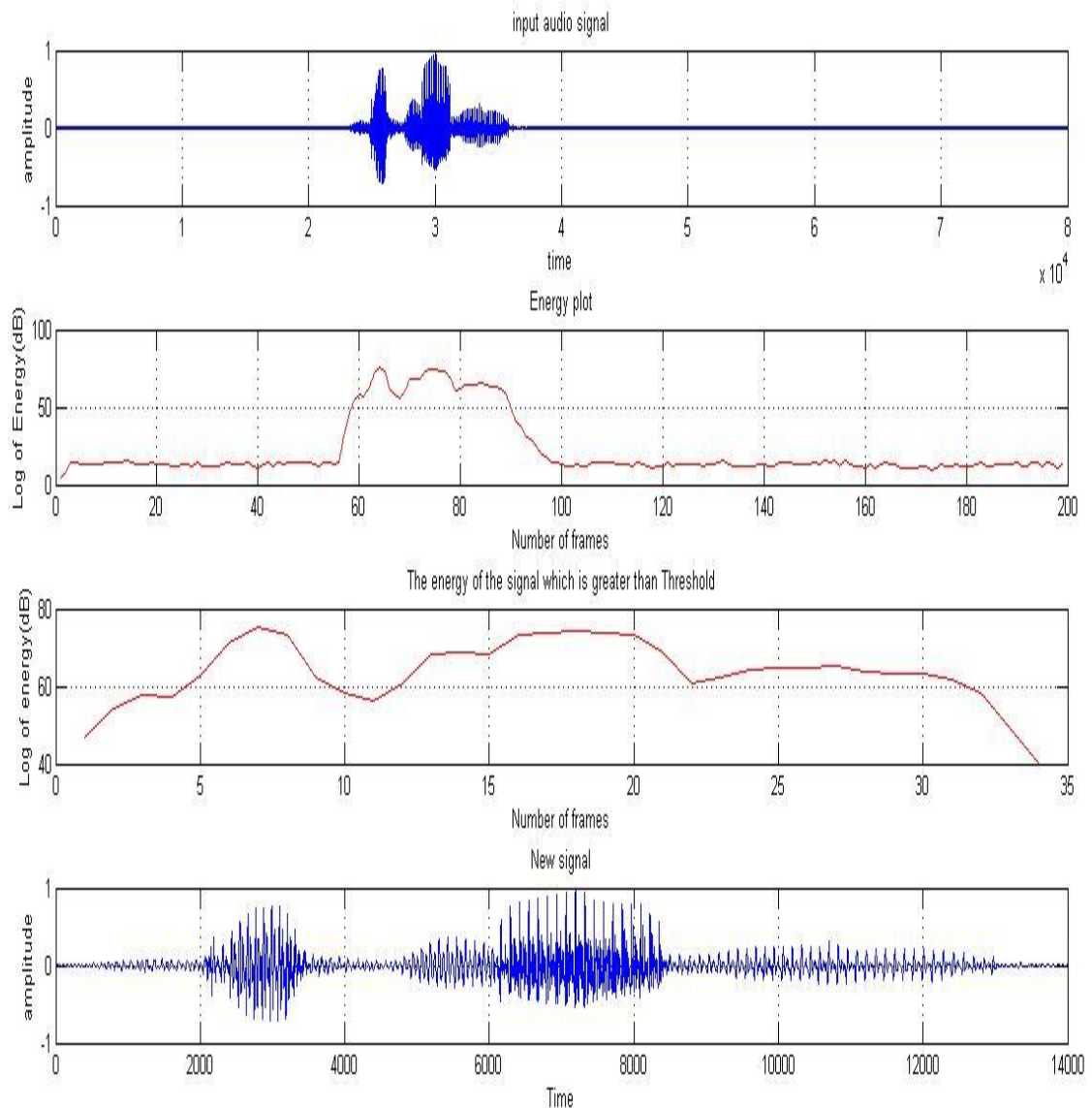


Fig 2.4 End point detection process waveforms

CHAPTER-3

FEATURE EXTRACTION

3.1 Introduction:

Feature extraction is an integral part of real time speech recognition system. The performance, quality and accuracy of speech recognition suffer greatly due to increase in background noises. Thus feature extraction is a technique to remove the changeability of the input speech signal while retaining the necessary characteristics of the speech. The speech signal is converted into useful parametric- representation, which can be further analyzed and classified. This process removes unwanted and redundant information and retains vital information. The main goal of feature extraction is to produce perpetually meaningful representation of digitalized waveform. It changes the input signal into acoustically identifiable components and keeps computations feasible.

3.2 Feature Extraction Using MFCC

In case of designing real time speech recognition system, extracting and selecting the most appropriate parameter in the speech signal and representation of it is an important task as it significantly affects the performance of recognition. Compact representations of a set of coefficients are done which are the results of a cosine transform of the real log of short-term energy spectrum expressed on a Mel-frequency scale. The use of about 20 MFCC coefficients is common in speech recognition, although 10-12 coefficients are often considered to be sufficient for coding speech.

Design of real time speech recognition system

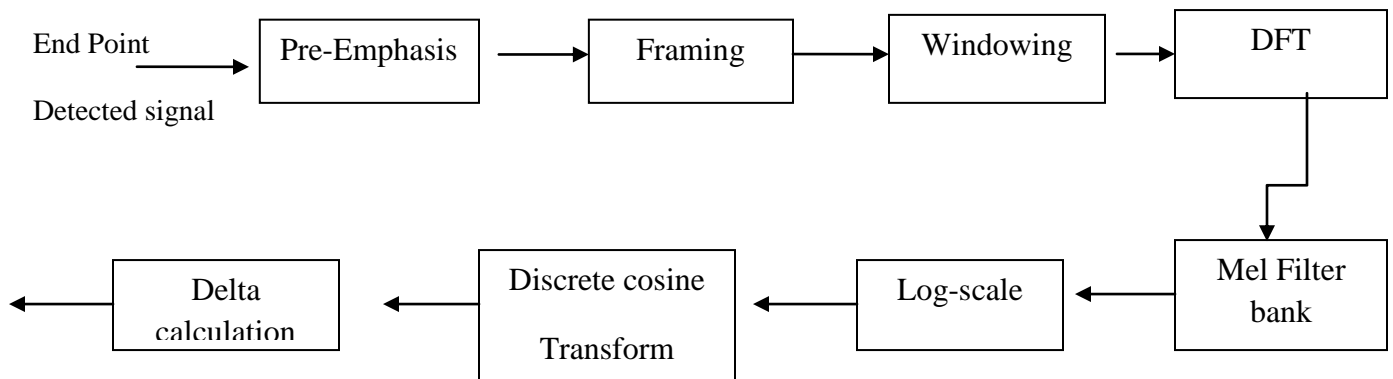


Fig 3.1 MFCC feature extraction algorithm.

a) **Pre-Emphasis:** Pre-emphasis refers to a system process designed to increase the magnitude of some higher frequencies with respect to the magnitude. In the lower frequency regions it improves the overall signal-to-noise ratio by minimizing the adverse effects of phenomenon such as attenuation distortion or saturation of recording media in subsequent parts of the system.

The speech signal represented by $x[n]$ is then sent to a high-pass filter as given in the below equation.

$$Y[n] = X[n] - \alpha X[n - 1]$$

Where $Y[n]$ is the output signal. The value of α is normally between 0.9 to 1.0.

The Z transform of this equation is given by:

$$H(z) = 1 - \frac{\alpha}{z - 1}$$

b) **Framing:** An audio signal is constantly changing, we assume that on short time scales the audio signal doesn't change much i.e. statistically stationary, the samples are constantly changing on even short time

scales. So in this project we have framed the signal into 25ms frames and 10ms of overlapping.

c) Windowing: It is a process in which the frames are multiplied with the hamming window

The Hamming window equation is given by

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

Now

$$f(n) = y(n) * w(n)$$

Where $f(n)$ represents frames

d) Discrete Fourier Transform (DFT): DFT is used to obtain the magnitude frequency response of each frame. It is considered that the signal within a frame is periodic and continuous when DFT is performed. If the signals are discontinuous even then it can be performed but the frame's first and last discontinuous points will cause undesirable effects in the frequency response. We can solve this problem by multiplying each frame with a hamming window to increase its continuity.

$$X[k] = \sum_{n=0}^{N-1} f[n] e^{-j\frac{2\pi n k}{N}}$$

e) Mel Filter Bank Processing: The speech signal does not follow the linear scale and the frequency range is outspread. The output is given by the sum of its filtered spectral components. Mel spaced filter bank is a set of 26 triangular filters that we apply to the period gram power spectral estimate.

The formula for converting from frequency to Mel scale is:


$$M(f) = 1127 * \ln\left(1 + \frac{f}{700}\right)$$

where f Represents frequency

To obtain frequency back from Mel scale we use:

$$M^{-1}(f) = 700 * [e^{m/1127} - 1]$$

MEL FILTER BANKS AND THEIR CORRESPONDING FREQUENCIES



FILTER NUMBER	CORRESPONDING FREQUENCY(Hz)	FILTER NUMBER	CORRESPONDING FREQUENCY(Hz)
01	300	15	2370.1
02	383.42	16	2626.3
03	473.79	17	2903.03
04	571.71	18	3204.4
05	677.80	19	3530.1
06	792.74	20	3882.9
07	917.27	21	4265.2
08	1052.2	22	4679.4
09	1198.3	23	5128.2
10	1356.7	24	5614.4
11	1528.3	25	6141.1
12	1714.2	26	6711.8
13	1915.6	27	7330.1
14	2133.7	28	8000




Fig 3.2 Mel Frequencies

Design of Mel filters Bank:

$$H_m(K) = \begin{cases} 0 & K < f(m-1) \\ \frac{K - f(m-1)}{f(m) - f(m-1)} & f(m-1) \leq K \leq f(m) \\ \frac{f(m+1) - K}{f(m+1) - f(m)} & f(m) \leq K \leq f(m+1) \\ 0 & K > f(m+1) \end{cases}$$

Where m is the number of filters we want, and $f(\cdot)$ is the list of $m+2$ Mel-spaced frequencies

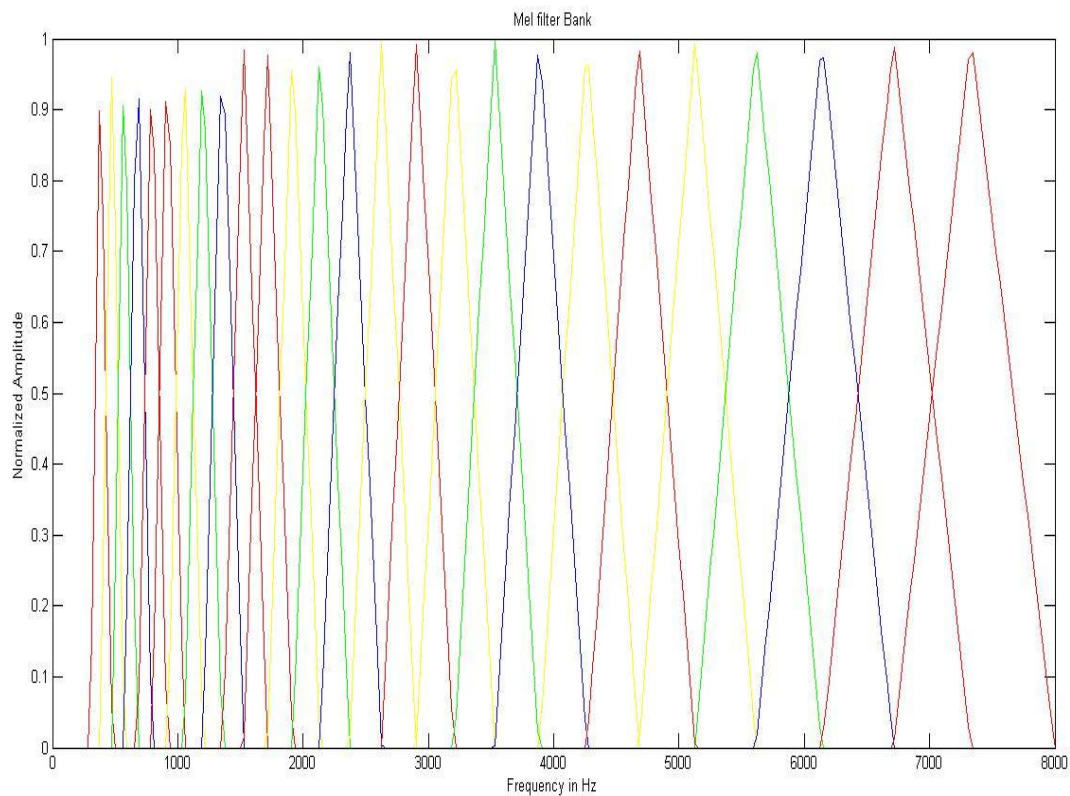


Figure 3.3 Mel filterbanks

f) Discrete cosine Transform (DCT): This process is used to convert from frequency domain into time domain using Discrete Cosine Transform (DCT). The result produces Mel Frequency Cepstrum Coefficients (MFCC). The representation of local spectral properties of the signal along with frame analysis is good when done by cepstral analysis. DCT is done using following equation:

$$Y[K] = \sum_{m=1}^M \log(y(m))^2 \cos(k(m - 0.5) \frac{\pi}{M})$$

Where $y(m)$ represents the log output of the Melfilterbank output.

In our project we have retained 2-13 co-efficients of DCT

g) Delta co-efficient calculation: In the speech signal frame changes mostly occur at its transitions. Therefore, more features must be added to make the cepstral analysis more efficient.

Delta coefficients are calculated using the formula

$$C^{t_i} = C^{t+1_i} - C^{t-1_i}$$

Where i varies from 1 to 26 and t represents frame number

We retain 12 delta co-efficient.

At last we have 25 Features as follows for each frame.

12 MFCC coefficients	12 Delta coefficients	1 Energy component
----------------------	-----------------------	--------------------

3.3 Filterbank Number as a Feature: This was another feature extraction technique which we tried. But the accuracy was low compared to the first one.

Procedure for this Feature extraction is as follows:

- Filter bank which has maximum power for each Frame is computed.
- For every frame if the power is above some threshold we retain the filter bank number.
- Else set the filter bank number to zero.
- Now this data is stored as a array of frames.
- We use this array as the reference matrix.

We recorded the digits from 0 to 9 and obtained the features for all the digits. Below figures shows the plot for few digits (3 samples for each digit). Plots with different colours represent different samples for each digit.

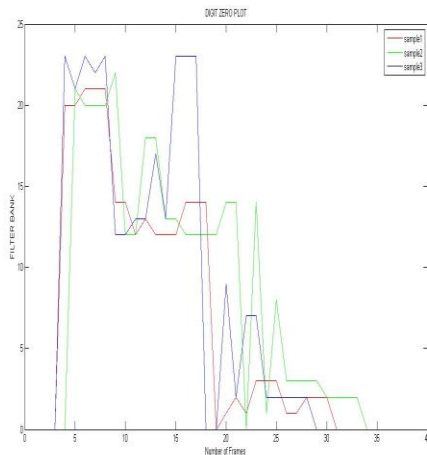


Fig 3.3 Plot for digit ZERO

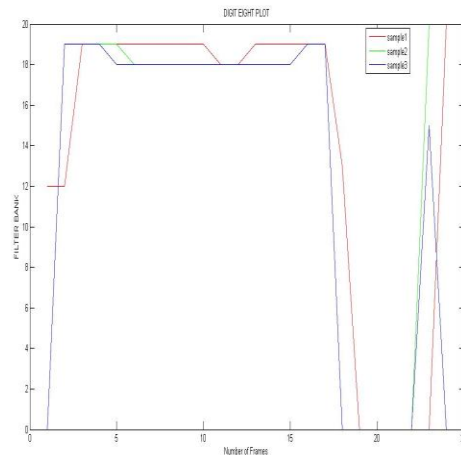


Fig 3.4 Plot for digit EIGHT

Design of real time speech recognition system

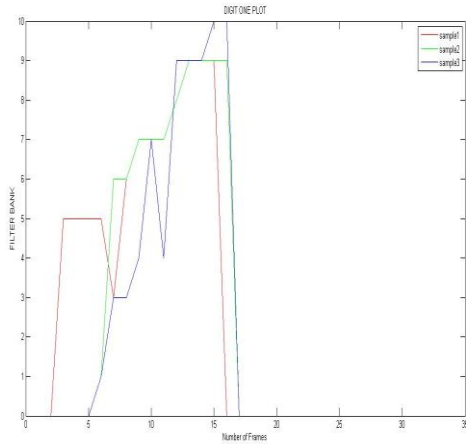


Fig 3.5 Plot for digit ONE

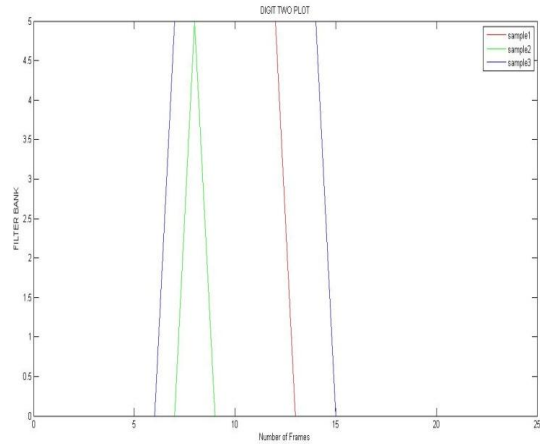


Fig 3.6 Plot for digit TWO

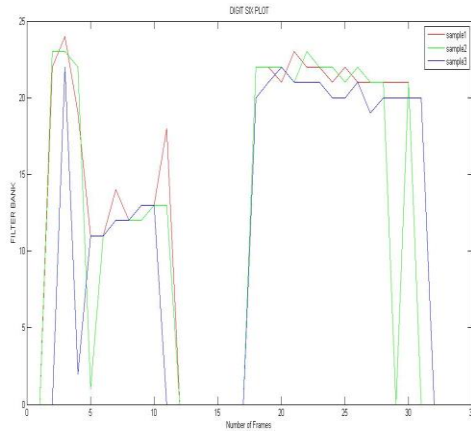


Fig 3.7 Plot for digit THREE

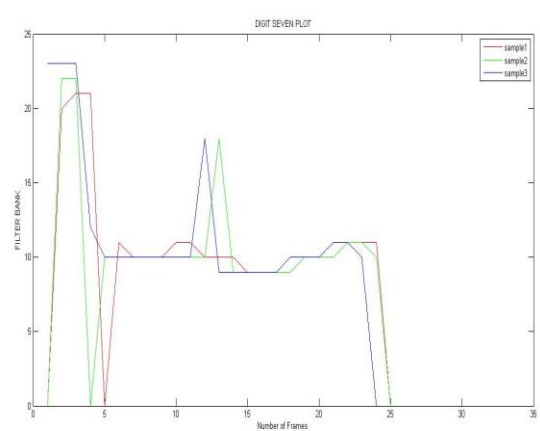


Fig 3.8 Plot for digit FOUR

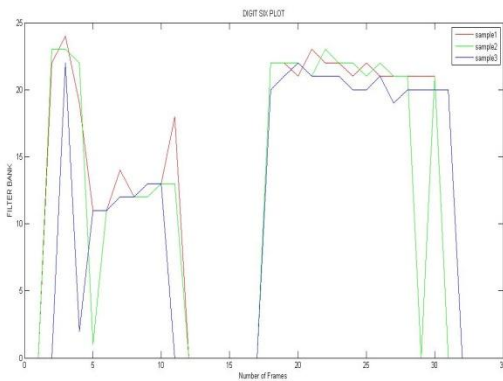


Fig 3.9 Plot for digit SIX

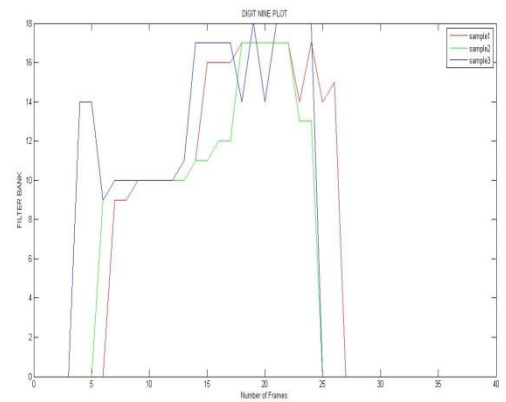


Fig 3.10 Plot for digit NINE

CHAPTER-4

FEATURE CLASSIFICATION

4.1 Feature classification using SVM:

Support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into higher dimensional feature spaces.

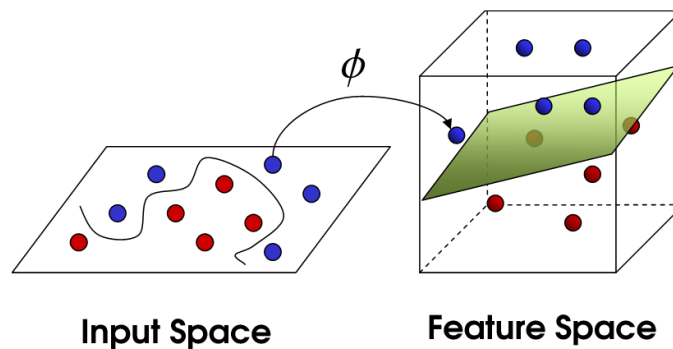


Fig 4.1 SVM Mapping

Radial basis function kernel, or RBF kernel, is a popular kernel function we have used in our project, 'c' & 'k' values were found to be 4 & 6 respectively.

$$g_i(x_j) = \exp\left(\frac{-\|x_j - \mu_i\|^2}{2\sigma_i^2}\right)$$

4.2 Feature classification using Euclidean classifier:

The minimum distance is calculated between the test speech signals with respect to the training dataset. A Euclidean distance measures the nearness between each training dataset to the test data. The Euclidean distance measure equation is given by

$$D_e(b_i, a_i) = \sqrt{\sum_{i=1}^N (b_i - a_i)^2}$$

The method is mathematically the simplest of all to find deviations between two datasets. But it is not the best one use.

CHAPTER-5

PHONEME EXTRACTION

5.1 Introduction:

Phonemes are abstract units and their pronunciation depends on contextual effects, speaker's characteristics, and emotions. During continuous speech, the articulator movements depend on the preceding and the following phonemes. The articulators are in different position depending on the preceding one and they are preparing to the following phoneme in advance. This causes some variations on how the individual phoneme is pronounced. These variations are called allophones which are the subset of phonemes and the effect is known as Co-articulation. For example In pronouncing digit 'ONE' we pronounce as WA_N

5.2 Phoneme Extraction procedures:

To extract the phonemes we have used wasp software and Matlab surf plots

a)The pre recorded audio signal is played in both wasp and matlab, in wasp we get corresponding frequencies plot for audio in matlab the output Mel filter bank represents the corresponding frequencies for the input audio signal

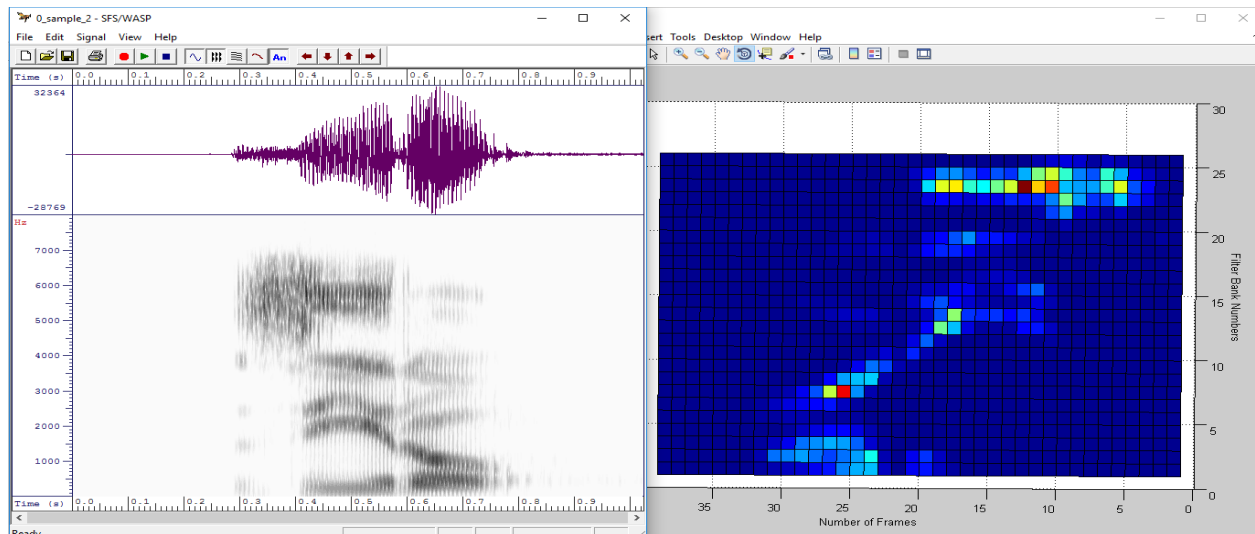


Fig 5.1 Phoneme Extraction for Digit ZERO

In the above figure left side plot is from the wasp software, where x-axis represents the time and the Y-axis represents the frequency in Hz. The right side plot is Mel filter bank output Y –axis represents the Filter bank number (we know each bank corresponds to range of frequencies) and X-axis represents the Frame numbers.

b) The small section of input audio signal is played in wasp software and corresponding frequencies are noted in the same way in matlab frames which corresponds to that frequencies are noted and we store 25 features like 12 MFCC, 12 Delta, 1 Energy of respective frames and we create Phoneme database.

c) For each phoneme we have 20 different samples & using all the phonemes samples we created Phoneme reference matrix which will be fed to first SVM classifier.

d) At the end of the first SVM classifier we get

Design of real time speech recognition system

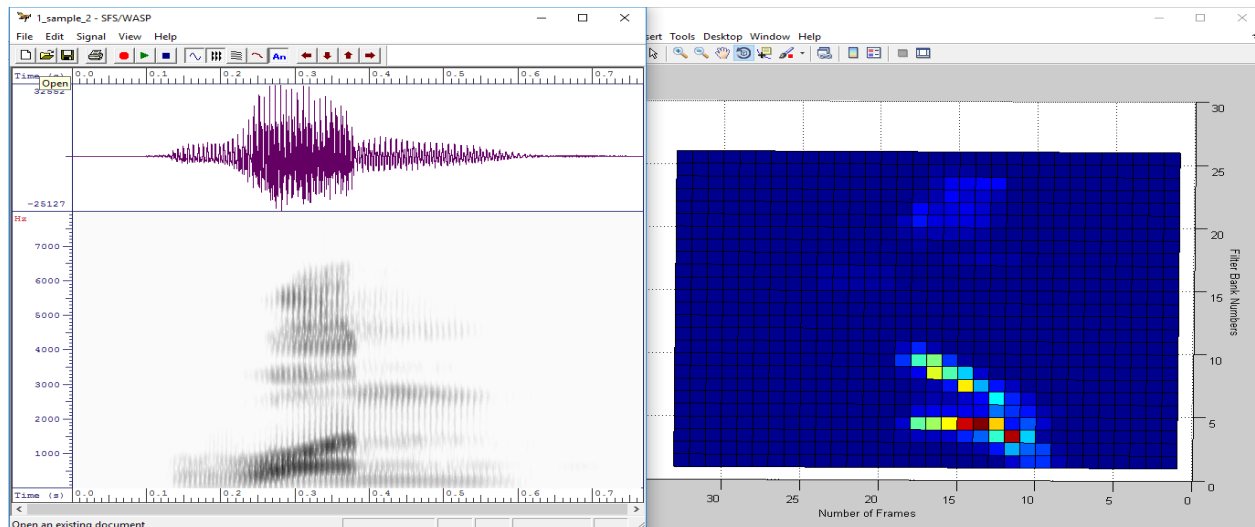


Fig 5.2 Phoneme Extraction for Digit One

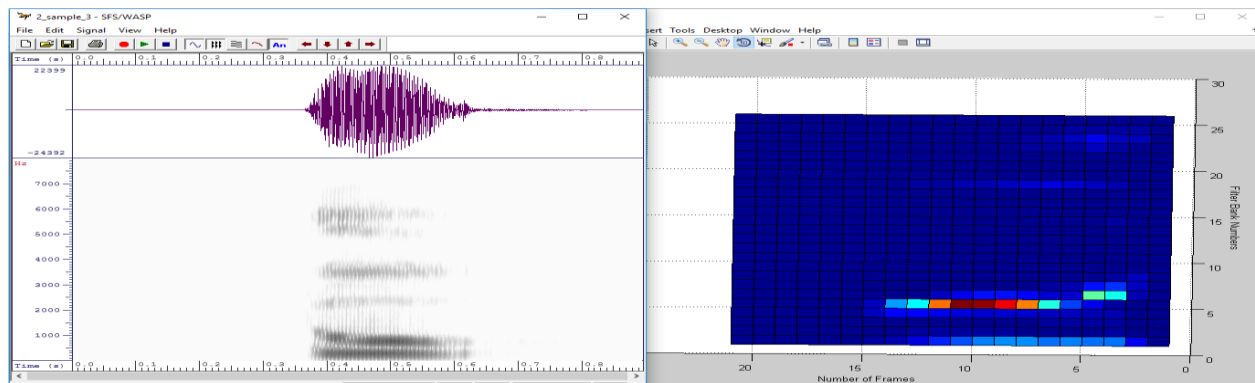


Fig 5.3 Phoneme Extraction for Digit Two

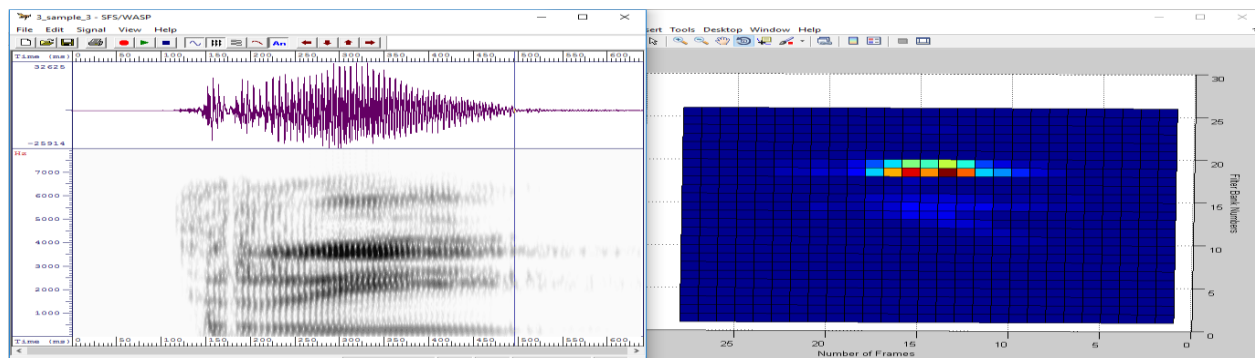


Fig 5.4 Phoneme Extraction for Digit Three

Design of real time speech recognition system

1	2	3	4	5	6	7	8	9	10
OW	OW	OW	F	OW	OW	OW	ZE	OW	ZE
ZE	OW	OW	ZE	ZE	ZE	ZE	ZE	ZE	ZE
ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE
ZE	ZE	ZE	SIH	ZE	ZE	SIH	ZE	ZE	SIH
ZE	ZE	ZE	K	ZE	ZE	ZE	ZE	ZE	ZE
ZE	SIH	ZE	ZE	ZE	S	ZE	ZE	ZE	ZE
ZE	SIH	SIH	IY	ZE	SIH	ZE	ZE	K	K
ZE	ZE	SIH	IY	S	ZE	K	ZE	IY	IY
ZE	ZE	ZE	IY	S	ZE	K	ZE	IY	IY
ZE	IY	ZE	K	K	K	K	IY	IY	IY
ZE	IY	K	IY	K	K	K	IY	IY	IY
IY	IY	IY	IY	K	K	K	IY	IY	IY
IY	IY	IY	IY	K	S	K	IY	IY	IY
IY	IY	IY	IY	K	S	K	IY	IY	IY
IY	IY	IY	IY	EY	K	K	IY	IY	IY
IY	IY	IY	IY	IY	K	K	IY	IY	TH
IY	IY	IY	TH	EY	IY	IY	F	TH	TH
IY	IY	IY	TH	EY	IY	IY	TH	TH	N
IY	IY	IY	TH	IY	IY	IY	RO	RO	RO
TH	TH	IY	RO	IY	IY	IY	RO	RO	RO
TH	TH	IY	RO	IY	OW	TH	RO	RO	RO
RO	TH	IY	RO	IY	TH	N	RO	RO	RO
RO	RO	TH	OW	TH	TH	K	RO	RO	RO
RO	TH	TH	TO	N	N	N	RO	TO	RO
RO	TO	RO	TO	N	N	N	RO	OW	
RO	TO	RO	TO	RO	TO	N	RO	TH	
RO	TO	RO	OW	N	TO	TO	RO	OW	
TO	TO	OW	TO	TO	TO	TO	RO	SIL	
TO	TO	TO	TO	TO	TO	TO	RO		
OW	TO	TO		TO	TO	OW	OW		
WAN	TO	TO		TO	TO	TO	RO		
WAN	TH	OW		TO	TO	WAN			
F	TH	TH		OW	OW	TO			
	FA	OW		TO	TO	TO			
	FA	TH		TO	TO	TO			
	FA	TH		TO	TO	WAN			
		F		TO	WAN	WAN			
		F			F	OW			
		FA							

Fig 5.5 Phonemes pattern for digit Zero for ten Utterances

TWO

1	2	3	4	5	6	7	8	9	10
TO	TO	TO	TH	TO	TO	TO	TO	TO	TO
TO	TO	TO	N	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	K	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	OW	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	OW	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	OW	TO	TO	TO	TO	TO	TO	TO	TO
TO	TO	TO	TO	TO	TO	TO	TO	TO	TO
WAN	TO	TO	TO	TO	TO	TO	TO	TO	TO
TO	OW	TO	TO	TO	F	TO	TO	TO	TO
OW	TO	SIL	TO	TO	F	TO	TO	TO	TO
	TO	SIL	TO	SIL	OW	TO	TO	TO	TO
		F	TO	F	OW	F	TO	TO	TO
			TO		TO	TO	TO	TO	TO
			TO		TO	TO	TO	TO	TO
			TO		OW	TO	TO	TO	TO
						TO	OW	TO	TO
						F	TO	TO	TO
								TO	TO
									ZE
									TO
									TO
									TO

Fig 5.6 Phonemes pattern for digit Two for ten Utterances

e) These output from the first classifier is stored in an array and it will be helps to create the database for the digits.

Finally the first classifier will classify or differentiate the phonemes and in the second classifier we can able classify digits based on different pattern for different digits.

So we have extended this idea of speech recognition to simple home automation system where we trained for simple commands like,

BLUE_LIGHT_ON, BLUE_LIGHT_OFF, RED_LIGHT_ON, RED_LIGHT_OFF, FAN_ON, FAN_OFF, TIME, and HELLO.

5.3 Digit recognition using phonetic approach:

In our project the first step is to develop text to speech system so we tried doing with isolated digits

- Here we use two SVM classifiers. One classifier is used to classify the phonemes and second is for the detection of phonemes.
- A total of 17 phonemes are detected for digits 0 to 9.
- Our first classifier detects the phonemes from the speech.
- Second classifier then detects the digits based on the phonemes obtained.

SL NO	DIGIT	PHONEMES
01	ZERO	ZE_IY_RO
02	ONE	WAN
03	TWO	TO
04	THREE	TH_EY_IY
05	FOUR	F_TO/OW
06	FIVE	F_FA_IVE
07	SIX	SIK_K_SIL_S/SIH
08	SEVEN	SIH/S_EHV_(N)
09	EIGHT	EY_SIL
10	NINE	N_IVE_EY/IY

Fig 5.7 Phonemes for digits

CHAPTER-6

TRAINING AND TESTING

6.1 Training :

a) Filter bank power feature with Euclidean classifier:

Here we have used the Filter bank power feature where we store the filter bank number which has maximum power for a each Frame and we classify using Euclidean classifier. The accuracy of this method was very low.

b) Filter bank power feature with SVM classifier:

We store the filter bank number which has maximum power for a each Frame and we classify using Euclidean classifier. The accuracy of this method was comparatively better than first one but in case of real time it processing it was not optimum.

c) SVM classifier using MFCC and Delta coefficients:

In this method we have stored MFCC and delta coefficients and classified with SVM classifier the accuracy was best compare to first two.

6.2 Testing:

Testing is one of the important tasks in case of speech signal

a) We have created 2 reference matrices which contains 10 samples of each phoneme in that one matrix is used for the training and another is used for testing.

b) We tested with different values of 'c' and 'k' of SVM classifiers
Where c represents the mean and K represents the variance.

Accuracies of different approaches:

DIFFERENT APPROACH OF CLASSIFICATION	TESTING WITH PRE RECORDED DATA SETS	TESTING WITH REAL TIME SCENARIO
Euclidean Classifier using Filter banks	70%	65%
SVM classifier using Filter banks	75%	70%
SVM classifier using MFCC and Delta coefficients	90%	80%

CHAPTER-7

HARDWARE COMPONENTS & INTERFACING

7.1 HARDWARE COMPONENTS:

1)Arduino UNO

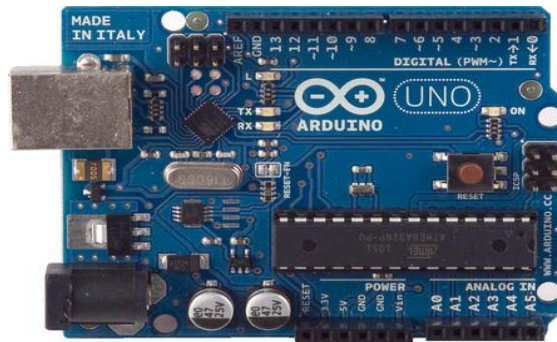


Fig 7.1 Arduino Uno

The Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega8U2 programmed as a USB-to-serial converter. We have used two Arduinos, one on the transmitter side and another on receiver side.

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Fig 7.2 Technical Specifications of an Arduino

2)RF Transmitter and Receiver

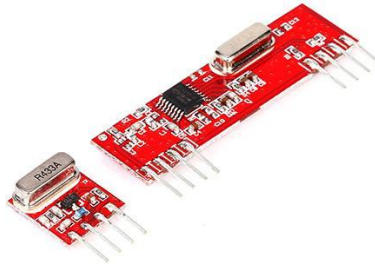


Fig 7.3 RF Module

The RF module operates at Radio Frequency. In this RF system, the digital data is represented as variations in the amplitude of carrier wave. This kind of modulation is known as Amplitude Shift Keying (ASK). This RF module comprises of an RF Transmitter and an RF Receiver. The transmitter/receiver (Tx/Rx) pair operates at a frequency of 434 MHz . An RF transmitter receives serial data and transmits it wirelessly through RF through its antenna connected at

pin4. The transmission occurs at the rate of 1Kbps - 10Kbps. The transmitted data is received by an RF receiver operating at the same frequency as that of the transmitter.

RF Pin Description:

Pin Description:

RF Transmitter

Pin No	Function	Name
1	Ground (0V)	Ground
2	Serial data input pin	Data
3	Supply voltage; 5V	Vcc
4	Antenna output pin	ANT

RF Receiver

Pin No	Function	Name
1	Ground (0V)	Ground
2	Serial data output pin	Data
3	Linear output pin; not connected	NC
4	Supply voltage; 5V	Vcc
5	Supply voltage; 5V	Vcc
6	Ground (0V)	Ground
7	Ground (0V)	Ground
8	Antenna input pin	ANT

Fig 7.4 RF module pin description

3) Light Emitting Diodes



Fig 7.5 LED

LEDs are a particular type of diode that converts electrical energy into light. We have used LEDs of two different colours. The LEDs are connected to the arduino on the receiver side.

4) DC Motor



Fig 7.6 DC Motor

A DC motor converts electrical energy into mechanical energy. We have used a DC motor in our home automation part to act as a Fan. It connected to the Arduino on the receiver side the just like the LEDs

5) L293D motor driver IC:



Fig 7.7 L293D IC

L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. L293D is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. We have used L293D IC to control the motor used for our application

6) Liquid Crystal Display:



Fig 7.8 Liquid Crystal Display

Liquid Crystal Display commonly called as LCD has the distinct advantage of having a low power consumption than the LED. A liquid crystal cell consists of a thin layer (about 10 μm) of a liquid crystal sandwiched between two glass sheets with transparent electrodes deposited on their inside faces. We have used a LCD to display the date and time for the home automation part of our project.

LCD Pin Information:

Pin	Symbol	Description	
1	V _{SS}	Ground	0 V
2	V _{CC}	Main power supply	+5 V
3	V _{EE}	Power supply to control contrast	Contrast adjustment by providing a variable resistor through V _{CC}
4	RS	Register Select	RS=0 to select Command Register RS=1 to select Data Register
5	R/W	Read/write	R/W=0 to write to the register R/W=1 to read from the register
6	EN	Enable	A high to low pulse (minimum 450ns wide) is given when data is sent to data pins
7	DB0	To display letters or numbers, their ASCII codes are sent to data pins (with RS=1). Also instruction command codes are sent to these pins.	8-bit data pins
8	DB1		
9	DB2		
10	DB3		
11	DB4		
12	DB5		
13	DB6		
14	DB7		
15	Led+	Backlight V _{CC}	+5 V
16	Led-	Backlight Ground	0 V

Fig 7.9 LCD pin info

RTC Module



Fig 7.10 RTC Module

This module keeps accurate time for years using a tiny coin-cell. A driver library allows your program to easily set or read the time and date. This module is connected to the Arduino on the receiver side with the LCD to display time and date.

RTC Pin Connections:

RTC pin	Arduino Pin
VCC	5V
GND	GND
SDA	A4
SCL	A5

7.2 Hardware Interfacing with Matlab:

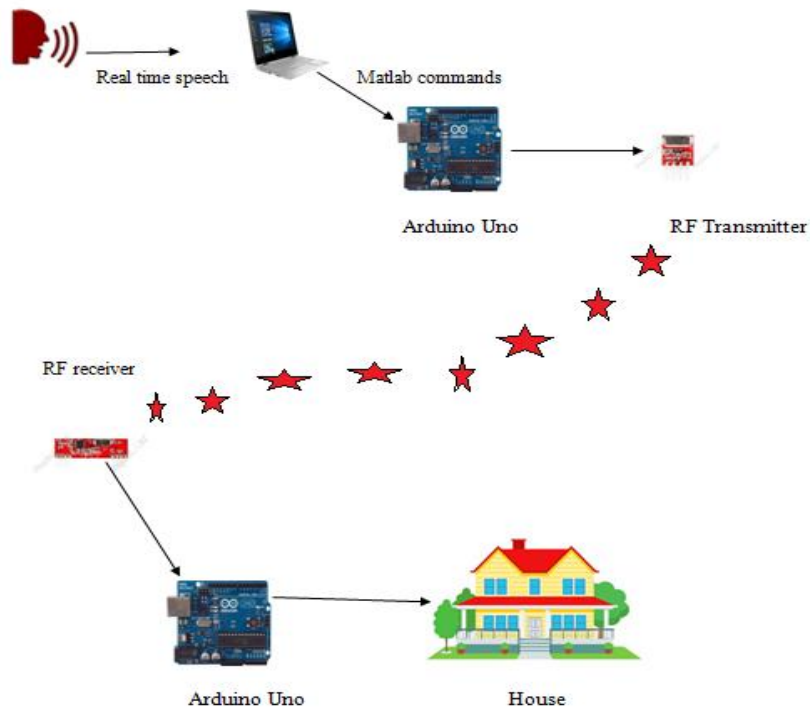


Fig 7.11 Hardware interfacing with matlab

The Interfacing part of our project has two divisions:

- I. The transmitter end
- II. The receiver end

The Transmitter End:

The transmitter end consists of a computer with the Matlab software installed in it. The Matlab is trained to accept the commands spoken by the user. The Matlab is interfaced with an Arduino microcontroller which is further connected to a RF transmitter. The Arduino is connected to one of the COM ports of the computer. Using that COM port a serial object is created with a baud rate of 9600. The command used here is

```
ser_obj=serial('COM7','BaudRate',9600);
```

We establish a connection between the Arduino and matlab using the command

```
fopen(ser_obj);
```

We have trained the system for 8 commands. Every Command is assigned with one unique code which is used by the RF transmitter to transmit using the Serial monitor. To transmit the code assigned over Serial monitor via matlab we use the command

```
fwrite(ser_obj,'1');
```

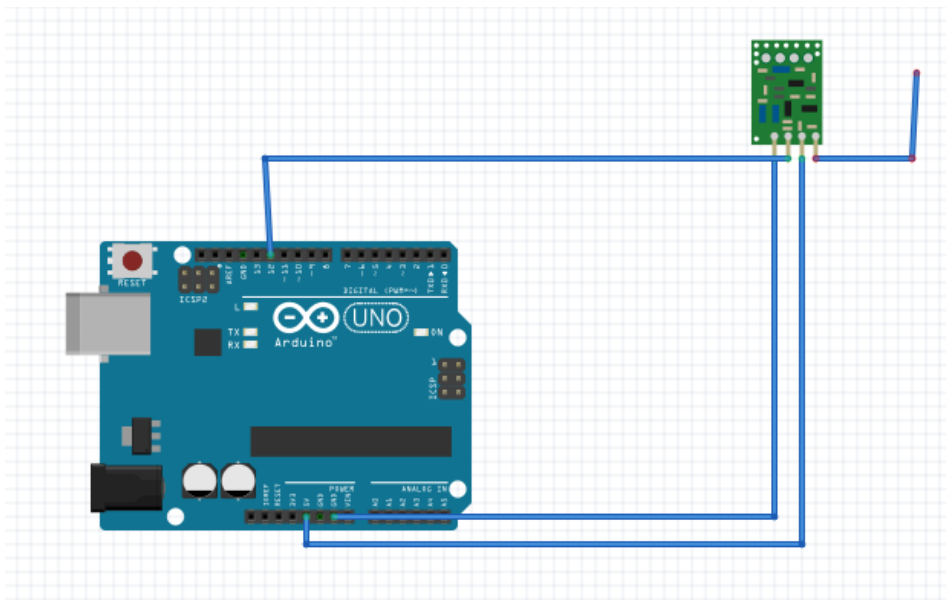


Fig 7.12 Interfacing of arduino with RF transmitter

The Receiver End:

At the receiver end we have used two RF receivers connected to two different Arduinos. One arduino is interfaced with LEDs and a DC motor while the other Arduino is connected to a RTC module and LCD .When a command is uttered by the user the code assigned to it is transmitted via transmitter. The

Design of real time speech recognition system

receiver on receiving the code performs the task assigned to it using Arduino. We have used LEDs of two different colours for two different rooms and a DC motor to act as a fan. A LCD display is used to display the command uttered and also the time and date when TIME command is spoken.

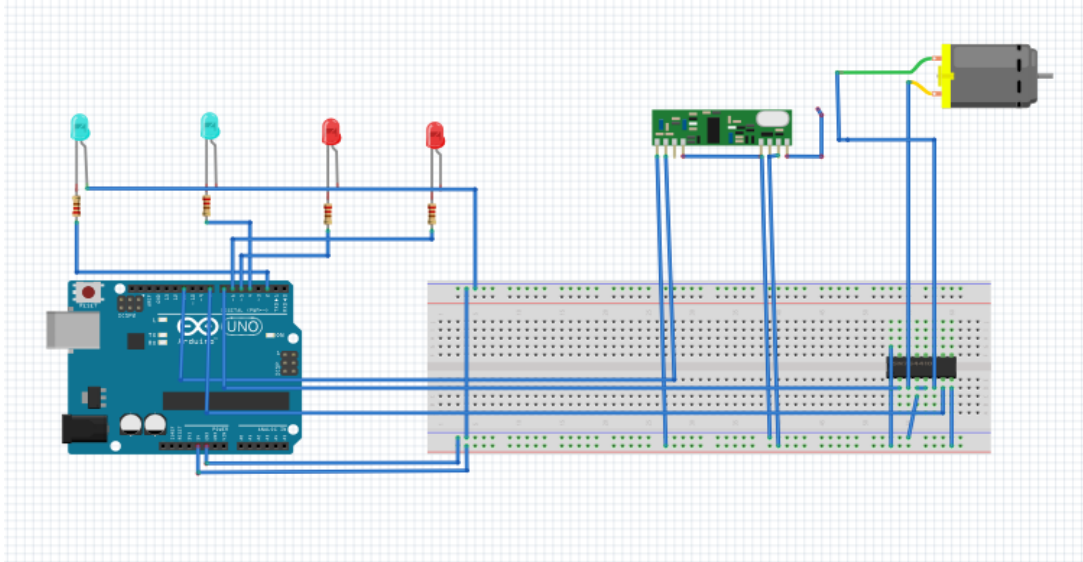


Fig 7.13 Interfacing diagram of the first Receiver

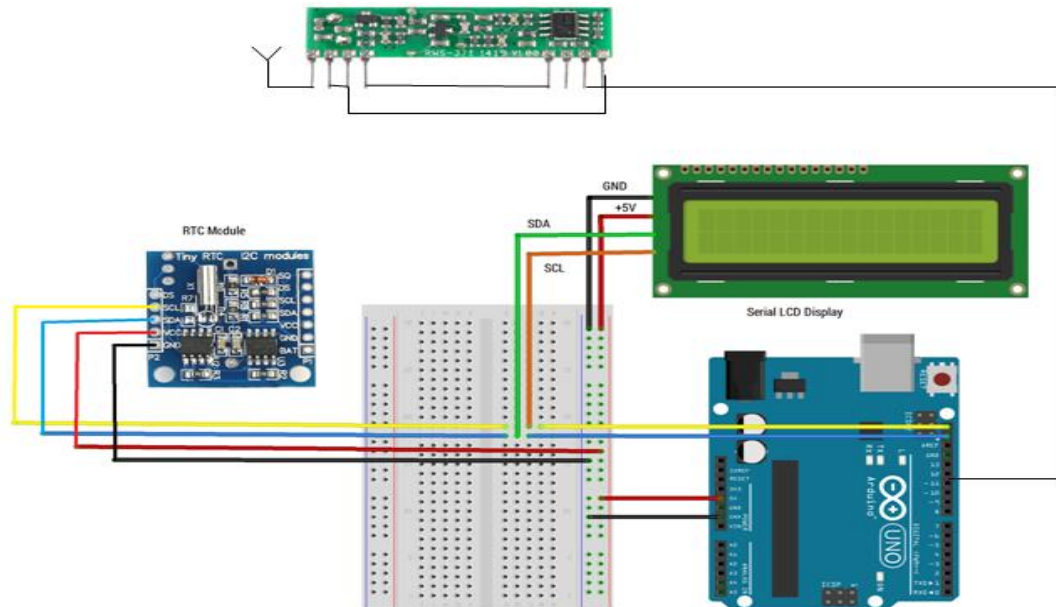


Fig 7.14 Interfacing diagram of the second Receiver

Commands and their respective Codes:

Command	Code
BLUE LIGHT ON	1
BLUE LIGHT OFF	2
RED LIGHT ON	3
RED LIGHT OFF	4
FAN ON	5
FAN OFF	6
GOOD MORNING	7
GOOD AFTERNOON	8
GOOD EVENING	9
GOOD NIGHT	a
TIME	b

Arduino Libraries Used:

1. VirtualWire.h (For RF Transmission)
2. LiquidCrystal.h (For LCD)
3. Wire.h (Communicate with I2C devices)

7.3.Graphical User Interface(GUI) Implementation:



Fig 7.15 GUI

Graphical User Interface was designed using matlab. The command "guide" was used to create a blank GUI where we could add push buttons and Command display box where we could see the command spoken by the user. Two push buttons were used. One to Start recording the signal and the other to process the signal.

CHAPTER-8

ADVANTAGES, DISADVANTAGES & APPLICATIONS

8.1 Advantages:

1) Increases productivity:

By speaking normally into the speech recognition system program, you create documents at the speed you can compose them in your head. People without strong typing skills or those who don't wish to be slowed down by manual input can use voice recognition software to dramatically reduce document creation time.

2) Can help with menial computer tasks, such as browsing and scrolling: People are becoming lazy day by day. They are not interested in doing the necessary routine work even. Previously there were punch cards to provide input to the system, then there came the keyboard, track ball, touch screen, mouse etc; all the previously used input methods require motion of hand or fingers. But, with speech recognition system user can provide input to the system through just his voice. He can complete most of his menial computer tasks easily.

3) Can help people with disabilities:

More recently students with physical disabilities have been able to use speech recognition system. Those with learning disabilities that affect their ability to write can now complete exams via voice recognition technology, and those with physical disabilities such as upper body paralysis can use speech recognition system to communicate effectively with others.

4) Diminishes spelling mistakes:

Even the most experienced typists will occasionally have spelling blunder; the average person is likely to make several mistakes in his or her composition. Speech recognition system always provides the correct

spelling of a word, thus eliminating the need to spend time running spell checkers.

8.2 Disadvantages:

1) Adaptability:

Speech recognition software are not capable of adapting to various changing conditions which include different microphone, background noise, new speaker, new task domain, new language even. Then efficiency of the software degrades drastically.

2) Out-of-Vocabulary (OOV) words:

Systems have to maintain a huge vocabulary of word of different language and sometimes according to the user phonetics also. They are not capable to adjust their vocabulary according to the change in users. Systems must have some method of detecting OOV words, and dealing with them in a sensible way.

3) Spontaneous Speech:

Systems are unable to recognize the speech properly when it contains disfluencies (filled pauses, false starts, hesitations, ungrammatical constructions etc). Spontaneous speech remains a problem.

4) Prosody:

Systems are unable to process Prosody (study of speech rhythms). Stress, intonation and rhythm convey important information for word recognition and users intentions (e.g., sarcasm, anger).

8.3 Speech recognition softwares:

1) Google Now

2) SIRI

3) Iris

4) Windows Speech recognition

8.4 Applications:

1) Games and edutainment:

Speech recognition offers game and edutainment developers the potential to bring their applications to a new level of play. With games, for example, traditional computer-based characters could evolve into characters that the user can actually talk to.

2) Speaker Identification:

Recognizing the patterns of speech of a various persons can be used to identify them separately. It can be used as a Biometric authentication system in which the user authenticates him/her self with the help of their speech. The various characteristics of speech which involves frequency, amplitude and other special features are captured with the previously stored database

3) Medical disabilities:

This technology is a great boon for blind and handicapped as they can utilize the speech recognition technology for various works. Those who are unable to operate the computer through keyboard and mouse can operate it with just their voice.

4) Fighter Aircrafts:

Pilots in fighter aircrafts have to keep a check on various fuctions going on in the aircraft. They have to provide a faster response to the sudden changes in the aircraft maneuver. They can give commands with their voice commands. It requires building a pilot voice template before. The actions are confirmed through visual or aural feedback.

8.5 Observations and Result:

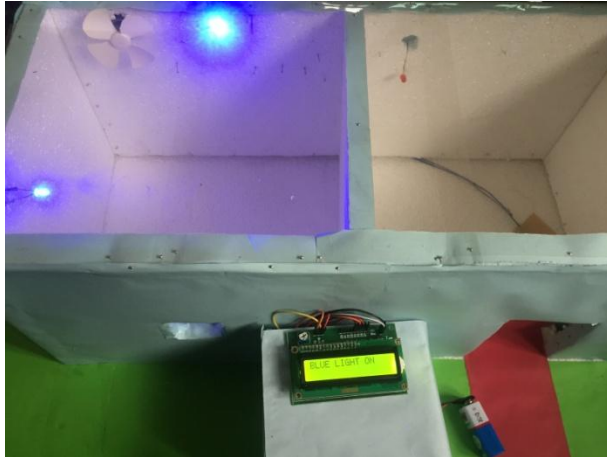


Fig 8.1 BLUE LIGHT ON

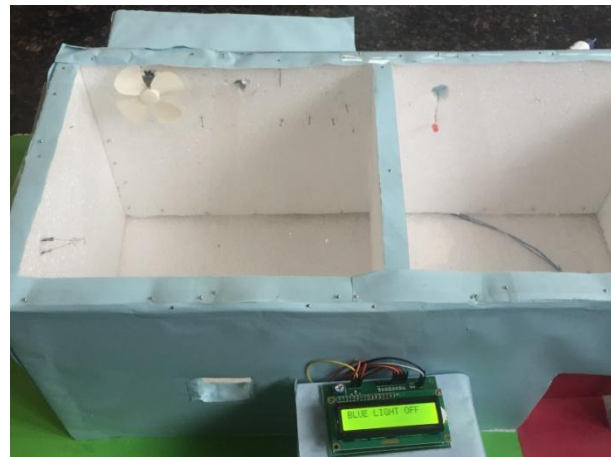


Fig 8.2 BLUE LIGHT OFF

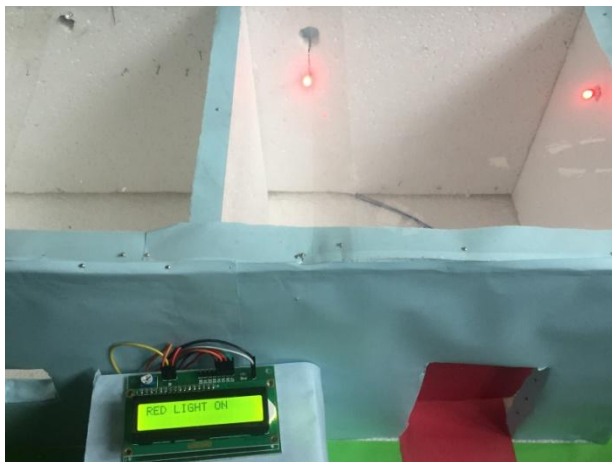


Fig 8.3 RED LIGHT ON



Fig 8.4 RED LIGHT OFF

Design of real time speech recognition system



Fig 8.5 FAN ON

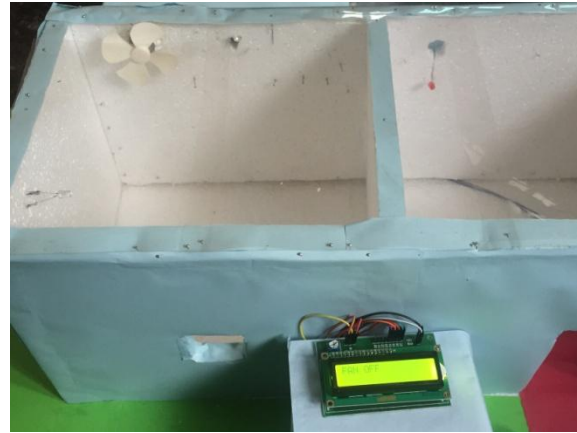


Fig 8.6 FAN OFF



Fig 8.7 HELLO



Fig 8.8 TIME

The above images show us the output observed when the respective commands are spoken by the user. The command spoken is also displayed on the LCD as well as on the GUI. When the HELLO command is spoken the system replies back with GOOD MORNING ,GOOD AFTERNOON and son on depending on the time of the day.

CHAPTER-9

CONCLUSION & FUTURE WORK

9.1 Conclusion:

Today, voice and natural language processing are at the forefront of any human machine interaction environment. We have designed a real time speech recognition system and developed one of its applications. While designing speech recognition system we studied and implemented feature extraction technique called Mel frequency cepstral coefficient. (MFCC). Then we executed 3 different feature classification techniques which are:

- 1) Euclidean classifier using filter banks
- 2) SVM classifier using filter banks
- 3) SVM classifier using MFCC, delta and power coefficients

The highest accuracy was obtained using SVM classifier using MFCC, delta and power coefficients. Later using this system was trained for digits from zero to nine. And next a home automation system was designed in which we could control the electrical appliances using speech. Here we have trained the home automation system for 8 commands. GUI was also designed for home automation.

9.2 Future Work:

- 1) We can train the system for more commands and could apply that for controlling actual appliances.
- 2) Making the speech recognition system speaker independent that will produce the same kind of output for a particular command irrespective of the user.

- 3) System which would be able to distinguish between nuances phrases and meaningful commands and would be able to process the proper command out of the nuances phrases correctly.
- 4) We could design an android or IOS app for controlling the devices
- 5) We can connect all the devices to internet for controlling them from anywhere (IOT).
- 6) Try to increase the efficiency by implementing other techniques like neural network, Relevant vector machine etc.

APPENDIX A

Contribution from team members:

All the three members have worked almost equally in every part of the project but certain portions were specifically assigned to a member where he worked more compared to the other two.

Dhananjay Kumar K L (1PI13EC030)

- a) Mel Filter Bank feature Extraction.
- b) Phoneme extraction for Digits and Home-automation commands.
- c) Training the system for different commands.
- d) Feature classification using MFCC and Delta co-efficient.

Karan G Barhanpur (1PI13EC039)

- a) End-point detection algorithm.
- b) Implementation of Pre-emphasis, Framing, windowing in MFCC.
- c) Feature classification using SVM classifier for Mel filter bank number Feature.
- d) LCD, RTC interfacing with Arduino.
- e) Implementation of GUI.

Kiran Uday Pai (1PI13EC041)

- a) Design of Mel filters banks.
- b) Implementation of DCT, Energy, Delta computation Blocks.
- c) Feature classification using Euclidean classifier for MFCC and delta Co-efficient.
- d) Arduino & RF transmitter interfacing with Matlab.
- e) RF receiver, LED, DC Motor Interfacing with Arduino.

APPENDIX B

References:

- [1] L.R.Rabiner and M.R.Samur , *An algorithm for determining the endpoints in isolated utterances*, The Bell System Technical Journal, Vol. 54, No. 2, Feb. 1975, pp. 297-315.
- [2] Bhadragiri Jagan Mohan , Ramesh Babu N, *Speech Recognition using MFCC and DTW*, Advances in Electrical Engineering (ICAEE), 2014 International Conference on, 19 June 2014, IEEE.
- [3] Mohammad Salman Haleem , *Voice Controlled Automation System*, Proceedings of the 12th IEEE International Multitopic Conference, December 23-24, 2008.
- [4] "Tutorial for MFCC algorithm implementation"
<http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/>
- [5] "Interfacing of Arduino, LCD and RTC module"
<http://cyaninfinite.com/tutorials/rtc-module-with-serial-lcd display/>
- [6] "Interfacing RF module with Arduino"
<http://www.electronicshub.org/arduino-rf-transmitter-receiver-module/>