



Department of Computer science & Engineering

M.Tech CSE

CS566 Speech Processing

Project Report On

Speech Based Contact Search App using
HMM

Guided By:-

Prof. P. K. Das

Submitted By:-

Aviral Singh(224101010)

Jasmin Borad(224101025)

Abinash Kumar Ray(224101062)

Contents

1	ACKNOWLEDGEMENT	1
2	ABSTRACT	2
3	MOTIVATION	3
4	INTRODUCTION	4
5	PROPOSED METHODOLOGY	5
6	FLOW DIAGRAM OF APPLICATION	6
7	WORKING	7
7.1	MODULES IN APPLICATION:	7
7.1.1	LIVE TESTING	7
7.1.2	LIVE TRAINING:	8
8	USER INTERFACE	9
8.1	Initial UI	9
8.2	Form1 Page	10
8.3	Recording Console	11
8.4	User Details Page	12
8.5	Update Details Page	13
9	Results	14

1 ACKNOWLEDGEMENT

This project is being submitted as a requirement for course fulfilment of CS566 - Speech Processing. It is a pleasure to acknowledge our sense of gratitude to Prof. P.K. Das who guided us throughout the project work. His timely guidance and suggestions were encouraging. We would also like to thank Teaching Assistants who were always helpful in clearing doubts. Finally, we thank to our classmates for the support.

1. **Aviral Singh (224101010)**
2. **Jasmin Borad (224101025)**
3. **Abinash Kumar Ray (224101062)**

2 ABSTRACT

Language is man's most important means of communication and speech its primary medium. Spoken interaction both between human interlocutors and between humans and machines is inescapably embedded in the laws and conditions of Communication, which comprise the encoding and decoding of meaning as well as the mere transmission of messages over an acoustical channel. Here we deal with this interaction between the man and machine through synthesis and recognition applications. Speech recognition, involves capturing and digitizing the sound waves, converting them to basic language units or phonemes, constructing words from phonemes, and contextually analyzing the words to ensure correct spelling for words that sound alike. Speech Recognition is the ability of a computer to recognize general, naturally flowing utterances from a wide variety of users. It recognizes the caller's answers to move along the flow of the call. Emphasis is given on the modeling of speech units and grammar on the basis of Hidden Markov Model. Speech Recognition allows you to provide input to an application with your voice. The applications and limitations on this subject enlighten the impact of speech processing in our modern technical field. While there is still much room for improvement, current speech recognition systems have remarkable performance. We are only humans, but as we develop this technology and build remarkable changes we attain certain achievements. Rather than asking what is still deficient, we ask instead what should be done to make it efficient.

The application is designed using C/C++ which listens to the user who speaks the name of a person within few seconds and the application displays the contact details of the spoken person's name which also includes his/her contact number, photo, address and email address.

3 MOTIVATION

The main objective behind making this project was to be able to help the people who can not see or the people who are not able to use their hands or the people who wants fast contact details with the help of their own speech which is effortless process compare to other processes and they also do not need to make any kind of screen contact while speaking which is itself an interesting way of getting the work done. Phone calls are the essential need of humans in every fields and day-to-day life. so, the easiness may come in people's life because of this project is unimaginable and as the technology advances the speech will play big roles in this kind of people.

4 INTRODUCTION

What is Speech Recognition:-

Speech Recognition is a technique which is quite popular now-a-days. When we speak into a micro- phone which is connected to the computer/mobile, it converts it to a text file which contains some amplitude values. Those values are basically the deviation of the speech signal from X-axis. Then we can use this file, do some calculations which can detect which word has been spoken and then further steps can be taken as per the requirement. One such example of application is Alexa. This report focuses on interactive speech recognition programs between humans and computers. The idea is to use the sounds of phonetics of speech to distinguish between the names, for example I have trained on names which can be easily differentiated by the model, like the name “yash” is recognized by the “shhh...” part of the name, “Ojas” is detected by the “Ooo” and “asssh” sound which differentiate these names easily from other names like “Ram” which itself is distinguishable as a whole, names “Riya” and “Amit” uses the vowel sound “e” at start but differs at the end. The names “david” and “Manoj” can be differentiated easily in speech using correct model Thus, in the end it comes to how much of your sound and of the user in general has the model been trained on. This project is just a minute start in the speech recognition world, to get people started on voice recognition.

5 PROPOSED METHODOLOGY

Basic requirements to develop this project are as follows:-

- Windows Operating System
- Microsoft Visual Studio 2010
- Recording Module
- Cool Edit 2000

With the availability of the above software, we further proceed in modelling the logic. The prerequisites of the project are:-

- Basic IO operations on file
- Pre-processing on speech data
- Feature extraction
- Modelling of extracted features
- Enhancing Model

6 FLOW DIAGRAM OF APPLICATION

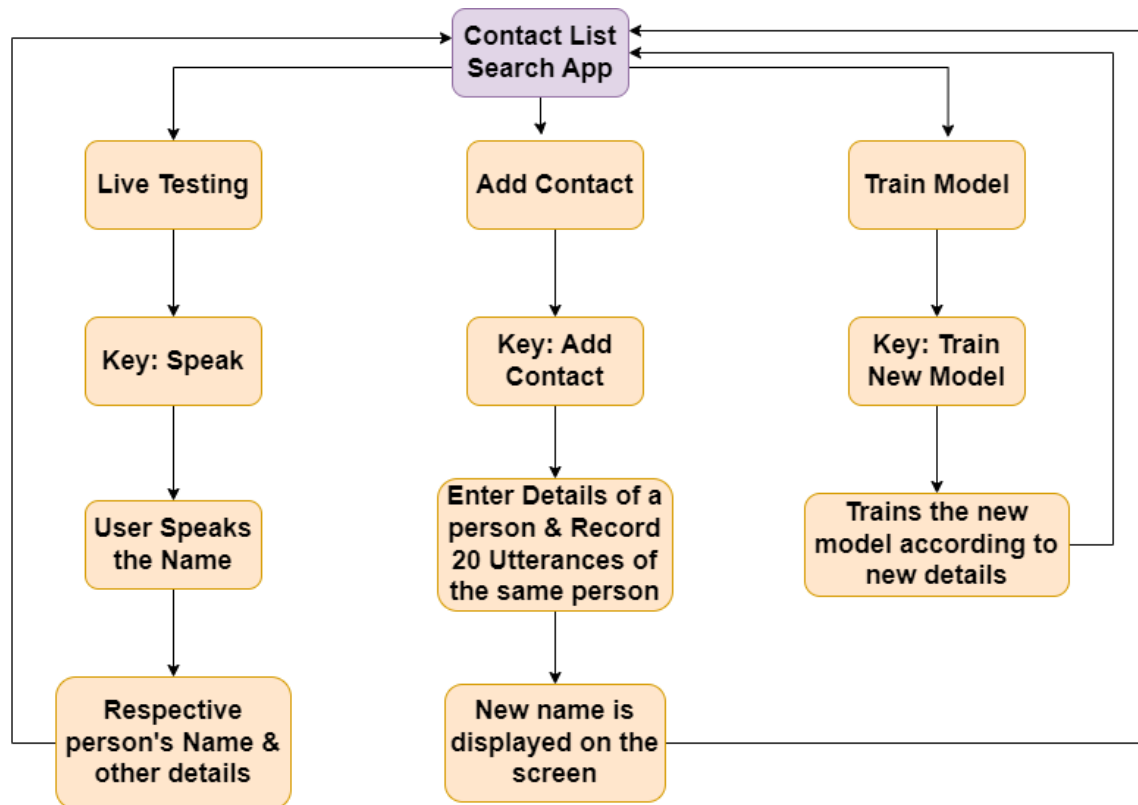


Figure 1: Flow Diagram

7 WORKING

The train, test data and models I have trained are stores based on the index of names given below (index, name) table:

Sr No	Contact Name
0	Ram
1	Vivek
2	amit
3	john
4	david
5	yash
6	manoj
7	ajay
8	ojas
9	riya

The names indexed 0 to 8 are already trained and their models stored in folder. Names indexed 9 is live trained from application and shown after training. Their samples are also stored in folder .

7.1 MODULES IN APPLICATION:

7.1.1 LIVE TESTING

The live recording.exe called which captures the user speech, preprocesses it, extracts observation sequence from codebook using nearest neighbour and checks with all trained models and returns index of model with highest $P(\text{Observation}/\text{model})$. This will be the model of uttered name.

7.1.2 LIVE TRAINING:

The training is called for remaining names (index 9) and they are trained. (After clicking add contact, wait for some secs), then the new trained name is shown. You can now live test for this name. All words are trained with 20 utterance each, so for your own files (if training for own data), make sure to provide 20 utterances of each word and the file names has to be same as used here.

Working of Training (of all names) :-

- Preprocess all names and extract Cepstral coefficients to generate a large Universe of Ci's.
- From the Ci Universe, call the LBG algorithm to generate a codebook (Vector Quantizer) of size 32.
- Now pass each name utterance to this codebook and get observation sequence using new name for each utterance.
- Create a base model (Feed Forward) to start the HMM.
- The model is bad(biased) , Now we train/teach it by passing data to recognize and differentiate it.
- The model is trained till it can't converge more and stored with respective name index.

8 USER INTERFACE

The first interface after running the project is as below which contains two window , one is Form1 and the another is recording console

8.1 Initial UI

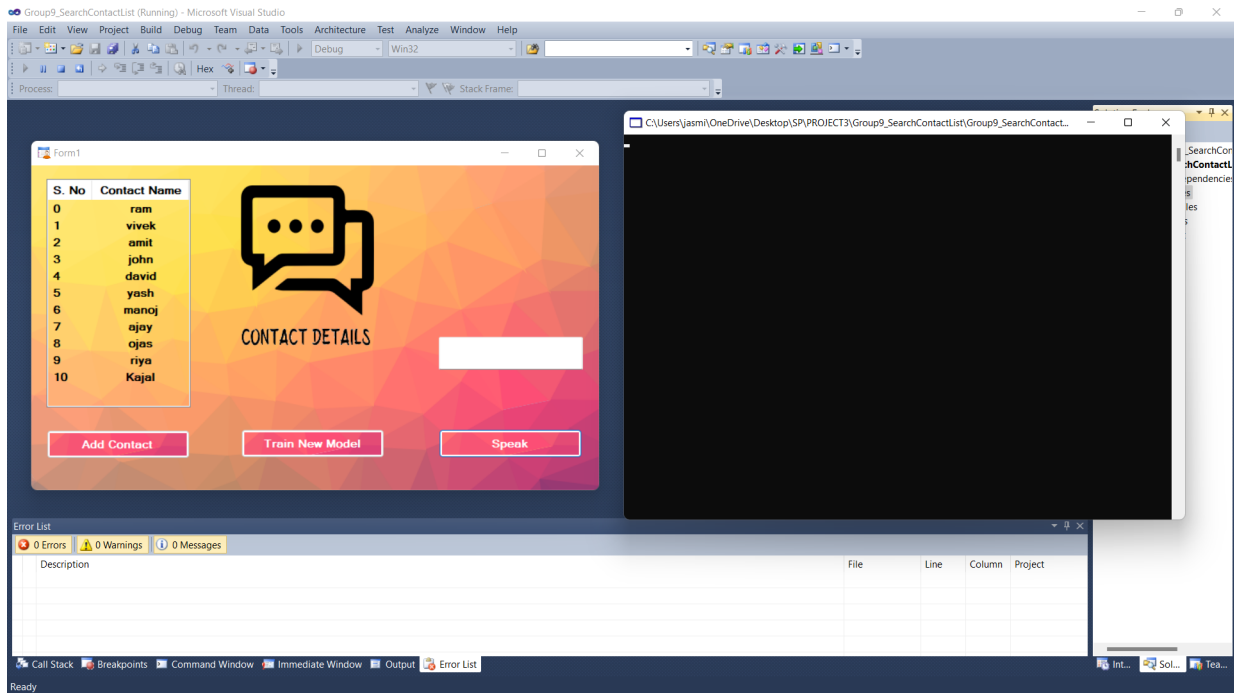


Figure 2: User Interface

8.2 Form1 Page

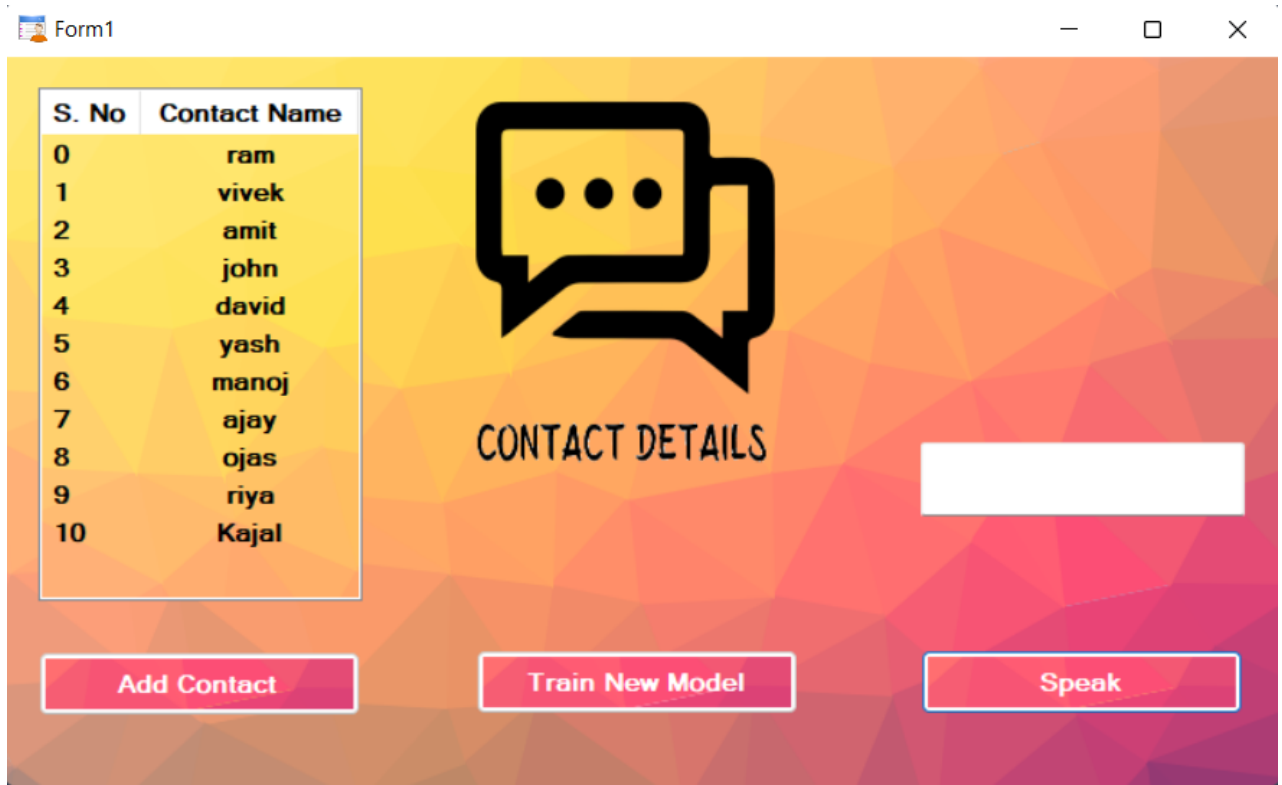


Figure 3: form1

This is the first interface with which user interacts directly which contains the options such as Add Contact, Train New Model and Speak which helps users in starting with speaking in order to get contact details. User can also add contact and train the model.

8.3 Recording Console

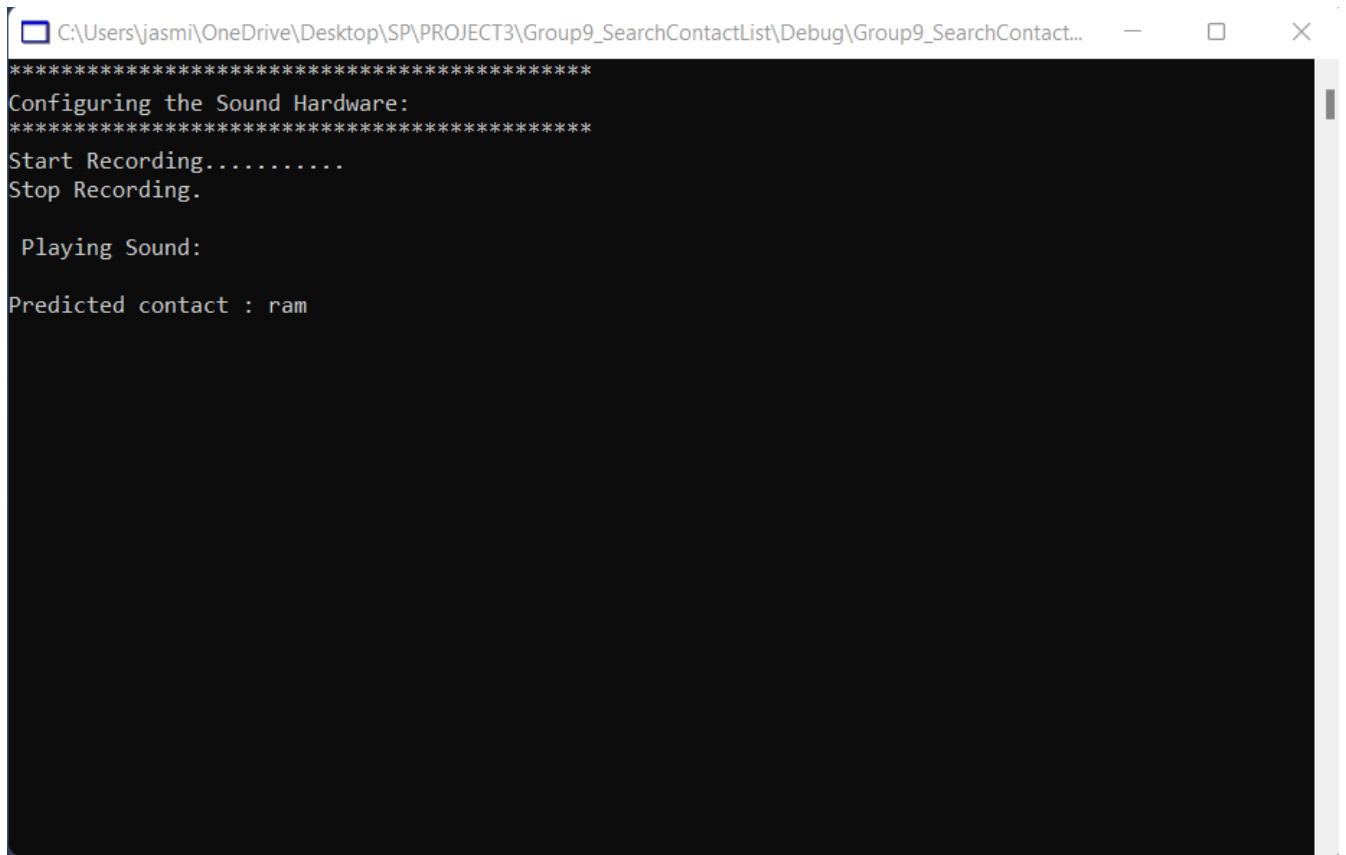


Figure 4: Recording Console

The Recording console is started recording once the user enters the speak button. The user get 3 secs to speak and once the user completes his/her speech in 3 secs then the sound will play whatever is recorded by the hardware and then the predicted word is displayed on the console as well as its contact details will be shown in another pop-up window.

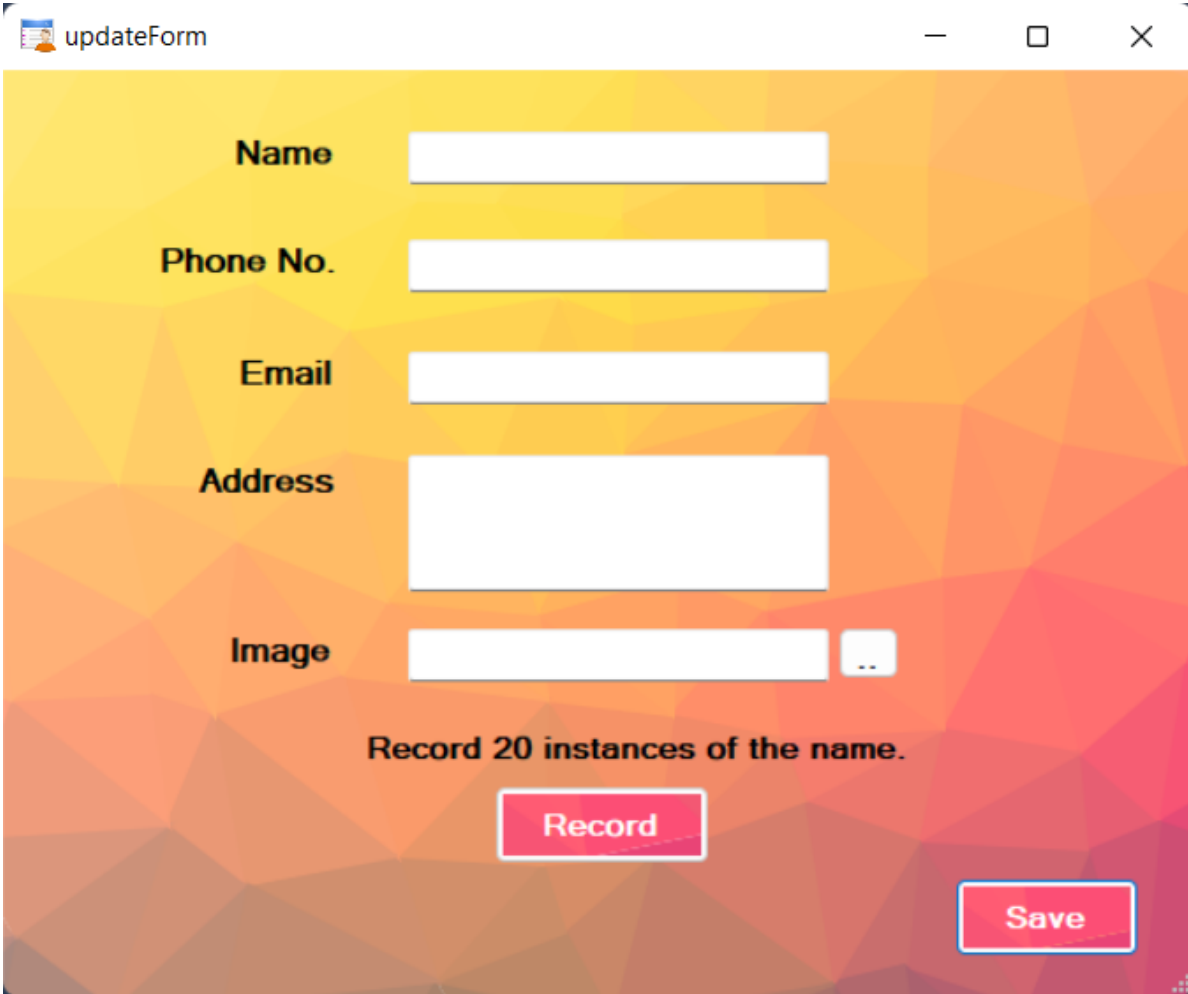
8.4 User Details Page



Figure 5: User Details Window

Once the user is identified by the software, its details will be shown in the pop-up window as follows and the option of closing the window is also provided in the interface of the window.

8.5 Update Details Page



The screenshot shows a window titled "updateForm" with a yellow and orange geometric background. The form contains the following fields and controls:

- Name**: A text input field.
- Phone No.**: A text input field.
- Email**: A text input field.
- Address**: A larger text input field.
- Image**: A text input field followed by a small square button with two dots.
- Record 20 instances of the name.**: A text instruction.
- Record**: A pink button with a white border.
- Save**: A pink button with a white border.

Figure 6: Update Details Page

The Add contact option is given in the starting window is also integrated with another window which will be shown on the screen once the user will enter the add contact button given in the window and user needs to fulfill the details of the user and subsequently will be required to record 20 utterances of the person and then finally save button is there to save the recordings and then press train new model option in order to train the new model.

9 Results

We are getting contact details of 10 people which are already stored in a data file. Fetching of data based on user's speech along with adding new user is successfully implemented.