International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 9 Issue 7 ǁ 2021 ǁ PP. 54-61

Human Computer Interaction through Morse Code

S. N. Deshpande, V. A. Deshmukh, G. D. Arjun, H. R. Goskonda, A. R. Butala, D. S. Datar
Department of Computer Engineering, P.E.S. Modern College Of Engineering, Pune

Abstract
The growth of technology in the medical field diminishes the difficulties of patients to an immense extent. The disease is named Motor Neuron Disease (MND). One of the major categories of physical disability leading to paralysis. MND patients are unable to do work like talk , walk, express feelings and communicate due to the weakening of muscles. The patient has control only upon his eye blinks, speech disorders are the most critical condition for human beings which causes brain damage, stroke to paralysis and several other diseases. It can also be caused from motor damage during accidents and leave a person completely unable to communicate. The only ability to remain intact in patients suffering from speech disorder and paralysis is control over eyelid movements. The patient has only a way to interact with others by eye movement only. By using this terminology, making a model where input will be the eyes blinking of paralyzed patients. To overcome the problem of communication (speaking or interaction) of paralyzed patients, the system will be built to implement such a communication model so that the patient will be able to communicate with the help of eyes blinking through morse code. So we are going to develop a computer vision application that is capable of detecting and counting the duration of eye blink in video streams on the basis of time span of eye blink, then the module will decide whether the blink is dash or dot. These dashes and dots sequence will be stored in an array then it will be decoded into normal text. Then further this text will be converted into audio by the pyttsx python library. In this way the message will be conveyed to another person by our system.
Keywords: *Morse Code, Eye Aspect Ratio (EAR), DLIB, OpenCV, Mouth Aspect Ratio(MAR)*

Date of Submission: 25-06-2021 Date of acceptance: 08-07-2021

I. INTRODUCTION

A paper is on human computer interaction for paralyzed people. Paralyzed people are unable to do hand movement or gesture communicating movement. It is also difficult for other people to communicate with paralyzed people. To communicate with the paralyzed people, one must learn the gesture communication method or refer to some book while communicating. This paper discusses the easy way of communication between paralyzed people and others with the help of computers.

Morse code is a method used in communication to encode text characters as standardized sequences of two different signal durations, called dots and dashes or dits and dahs. As paralyzed people are capable of eye movements, they can give signs like yes or no using eye blink. In our system, we have implemented human computer interaction through morse code. Further this morse code is converted into text and text to audio.

The growth of technology in the medical field diminishes the difficulties of patients to an immense extent. The disease is named Motor Neuron Disease (MND) .One of the major categories of physical disability leading to paralysis. MND patients are unable to do work like talk, walk, express feelings and communicate due to the weakening of muscles. The patient has control only upon his eye blinks. Speech disorders are the most critical condition for human beings which causes brain damage, stroke to paralysis and several other diseases. It can also cause motor damage during accidents and leave a person completely unable to communicate. The only ability to remain intact in patients suffering from speech disorder and paralysis is control over eyelid movements. The patient has only a way to interact with others by eye movement only. By using this terminology making a model where input will be an eye blink of paralyzed patients.

1.1 Aim & Objectives

To create a system for human computer interaction using morse code technique and for this calculate eye blink duration with the help of eye aspect ratio to convey information or message to the other person as audio output.

1.2 Requirements

The user interfaces required are as follows:-

1. Software Developers: People with very good knowledge of programming language projects, in order to understand and be able to extend the project's source code.
2. Helper or Technician : People with basic knowledge of technology for setup , login , registration procedure.

3. End User: People with disabilities, people who cannot move their lower body and cannot speak.

The hardware requirements are Processor-i3, Ram-4GB, HDD-1TB, Web Camera with resolution 720 p. The software requirements are Anaconda Navigator, Python Tkinter, Pyttsx. User Face should be clearly visible and video quality should be clear. Users should know the time span of morse code and they should have the knowledge of the morse code.

1.3 System Design

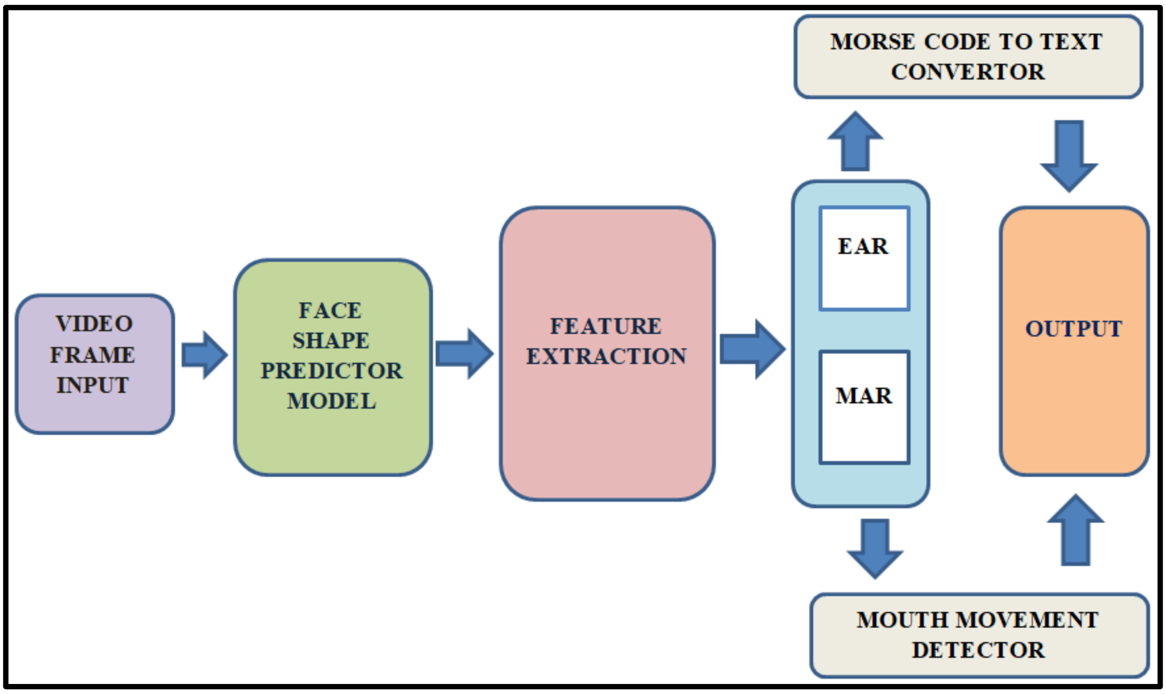

Figure 1: System Architecture

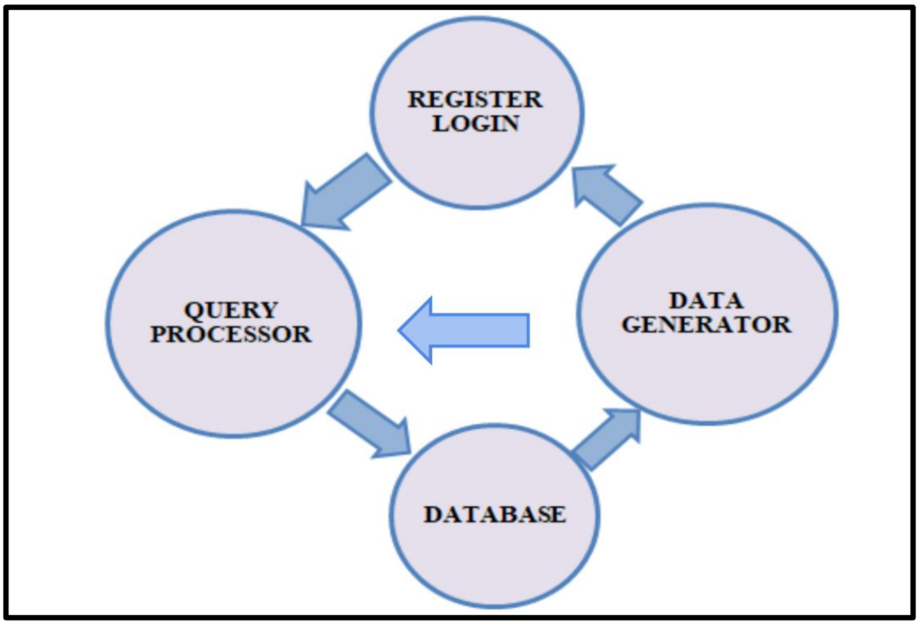

Figure 2: Frontend Architecture

Step 1: Face and Eye Detection
We use the Dlib library which is a pre-trained machine learning model for face detection and landmark plotting. We apply facial landmark detection to localize important regions of the face, including eyes, eyebrows, nose, ears, and mouth, we can extract specific facial structures by knowing the indexes of the particular face parts.

Step 2: Eye aspect ratio calculation

Each eye landmark is constructed using 6 points x,y-coordinates. We take it as P1-P6, the eye aspect ratio is the distance from P1, P4 to distance from (P2, P2) and (P5, P6). Indirectly we are finding the horizontal and vertical distance of an eye. The eye aspect ratio when the eye is opened is greater than 0.25. When eye is closing the eye aspect ratio decreases gradually and comes near to zero and all the points plotted on the eye form a straight line when the eye is closed. The equation which is used to find eye aspect ratio as follows,

$$\boxed{\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}}$$

Figure 3: EAR Formula

Input : P1 ,P2,P3 ,P4,P5,P6 landmarks on eye with (X,Y) coordinate.
Horizontal distance : H1 = P1-P4
Vertical Distance : V1 = P2-P6 , V2 = P3 -P5
EYE ASPECT RATIO = — V1 — + — V2 — / 2 — H1 —

Step 3 : In this we calculate the duration of eye blink for closed eye aspect ratio value. We set the threshold value of EAR. We calculate the duration of eye blink when EAR value is below the threshold value.

Step 4 : On the basis of duration of eye blink the input is taken as dots or dashes. The duration of an eye blink is calculated using EAR values, these values are analyzed by the system and decide whether the input taken is dot or dashes.

Step 5 : Dot-Dash sequence measure code is converted into normal text. Dot - Dash sequence is stored in an array. This sequence is converted into the respective letter.

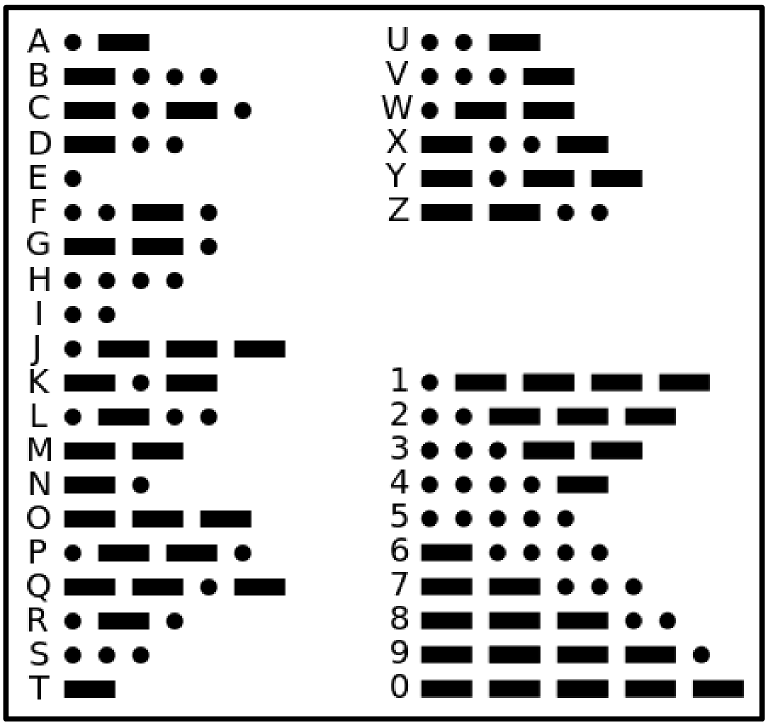

Figure 4: Morse Code

Step 6 : There are several APIs available to convert text to speech in python. Pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3. First mouth aspect ratio (MAR) is calculated. After detection of the mouth movement the text is converted into sound.

$$\boxed{MAR = \frac{\| M2 - M8 \| + \| M3 - M7 \| + \| M4 - M6 \|}{3 \, \| M1 - M5 \|}}$$

Figure 5: MAR Formula

1.4 Project Implementation:

This project was undertaken to take the input from the user in the form of eye blink and to produce sound for that particular input. To do so, the project was divided into different modules. In the first module input is taken from the user. The shape predictor 68 model detects the facial landmarks. The model will detect the left eye, right eye and mouth points. The second module detects the eye blink and classifies it as dot or dash on the basis duration of eye blink which is calculated using EAR value. The third module converts the dot and dash format into text. The fourth module converts the text into sound using the pyttsx library.

• Facial Landmark Detection Algorithm:
To detect the facial features, the Dlib facial landmark detector was used. It is a pre-trained detector of 68(x, y)- coordinates trained on the iBUG 300-W dataset. Facial landmarks are detected by using the shape predictor 68 model which is created using the DLIB library. The trainer is a support vector machine (SVM) that returns the "best fit" hyperplane to categorize the set of images. This detector localizes 68 points on the face.
The steps involved:
Step 1: Localize the face in the image.
Step 2: Localize the points for the left eye, right eye and mouth.

• Eye Blink Detection Algorithm:
Each facial feature can be accessed by indexing. The eyes can be accessed through points [37, 42] and [43, 48] for the right and left eye respectively in the facial landmark dictionary inside one of Python's libraries, "imutils". To detect blinks in real time, we have used the eye aspect ratio (EAR) value. The EAR for each eye is calculated and they are averaged to give the EAR of both eyes blinking. If the eye is closed then the EAR value gradually comes near to 0.
The steps involved:
Step 1: Point feature extraction.
Step 2: Eye aspect ratio (EAR) calculation.
Step 3: Calculation of time duration.
Step 4: Based on the duration of eye blink the input is considered as dot or dash.

• Open Mouth Algorithm:
This algorithm detects open mouth and close mouth. The mouth can be accessed through points [61,68]. Using these points the mouth aspect ratio (MAR) is calculated. Based on the value of MAR, an open mouth or closed mouth is detected. On the mouth movement of the user the text is converted into sound. There are several APIs available to convert text to speech in python. Pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3.
The steps involved:
Step 1: Point feature extraction.
Step 2: Mouth aspect ratio (MAR) calculation.
Step 3: Based on the value of MAR, mouth movement is detected.
Step 4: Text is converted into sound.

1.5 Testing:

The system being designed only on one data source and the data extracted from these sources is done without any tools manually, no automated testing was performed. Black box testing: it will test the valid username and password to login to the right account. Shape-predictor Model accuracy is tested with MS-LSTM model accuracy. Manual testing on Database, data insertion, selection and connection is tested. The testing performed on the prepossessing algorithm revealed that it could handle different data sources efficiently. The front end was a basic GUI which contained login, registration and history pages.

Table 1: Test Cases

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Login	Valid user and password	login successfully	successful Login
T02	Login	Valid user and invalid password	Invalid user or Invalid password	Unable to login
T03	Morse code decoding	It should return valid letter of entered Morse code	return valid Morse code	Return valid Morse code
T04	Data entry and extraction from database	Enter word into history and get history	processed successfully	Every word added to database and it should reflect in history

II. RESULT AND DISCUSSION

Step 1: After running the application we get the first login page. User will have to first register himself/herself.

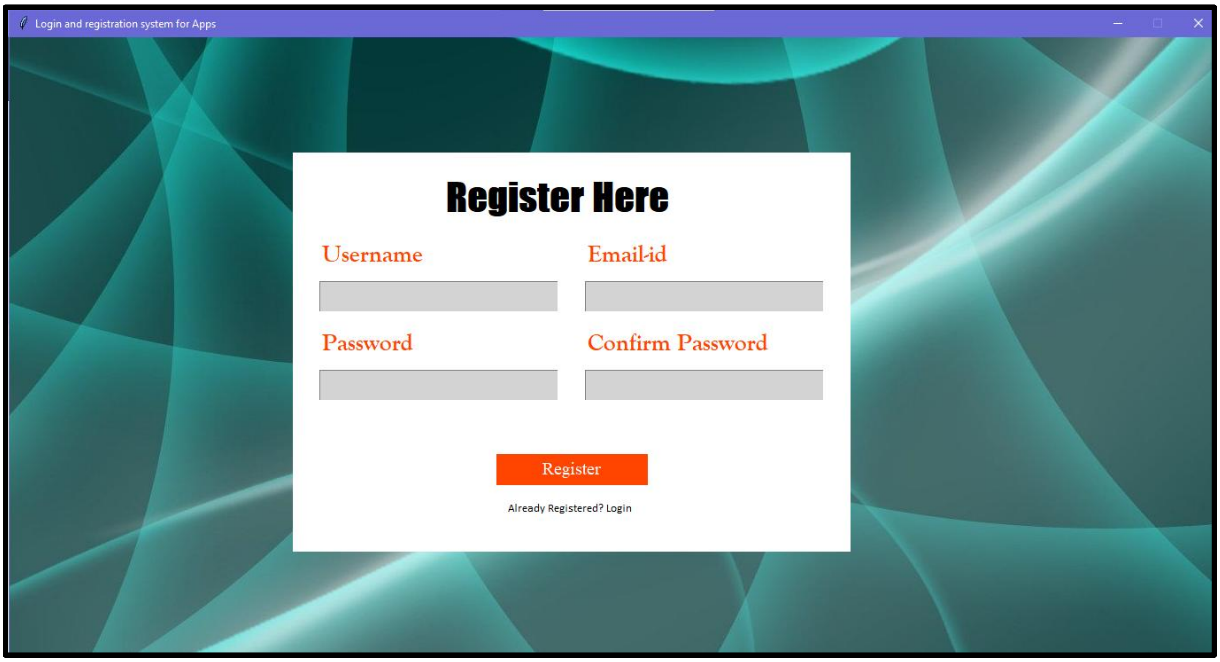

Figure 6: Registration Page

Step 2: After successful registration we will go to the login page using the login link. Login page consists of the user's email id, password and registration link. Users will login using his/her login credentials.

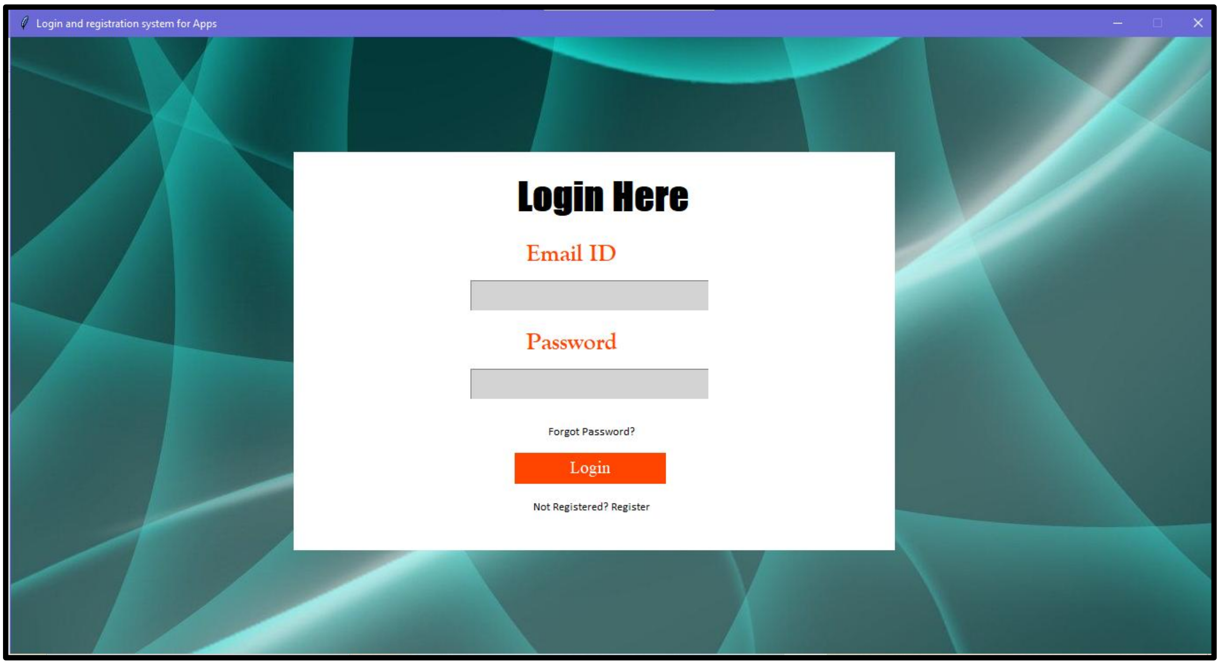

Figure 7: Login Page

Step 3: After successfully login the user will redirect into morse code system,it has three options Open Camera, History, Logout.

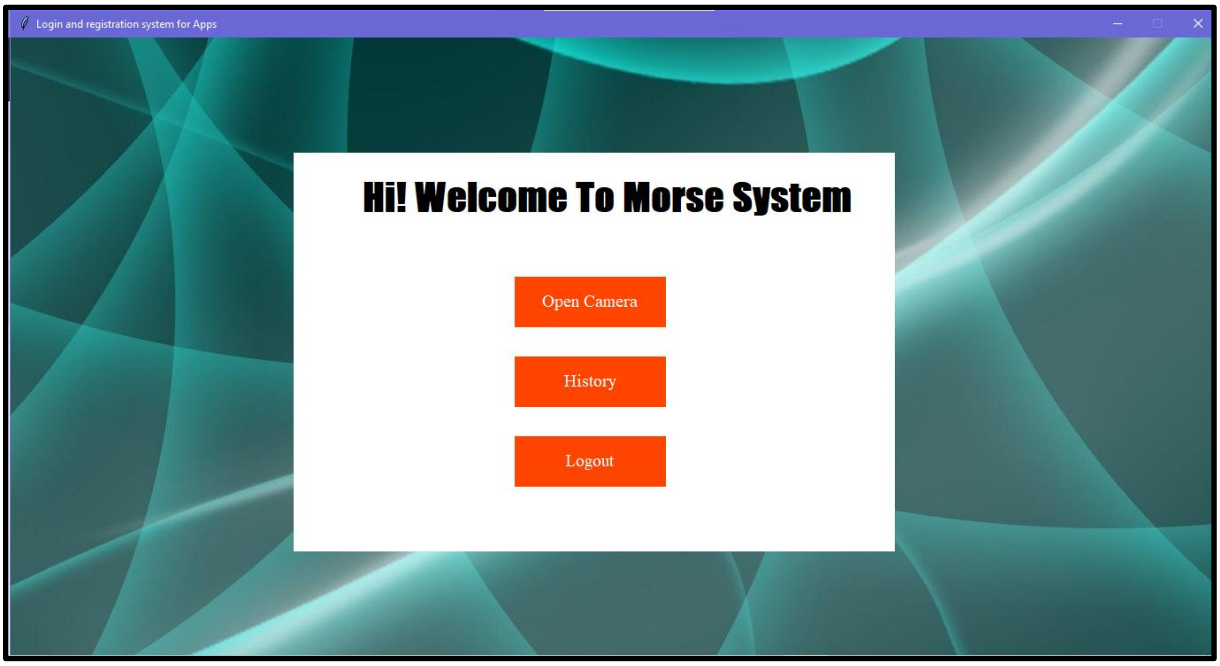

Figure 8: Morse Page

Step 4: After clicking on the open camera button the camera starts taking input from the user in the form of eye blink.

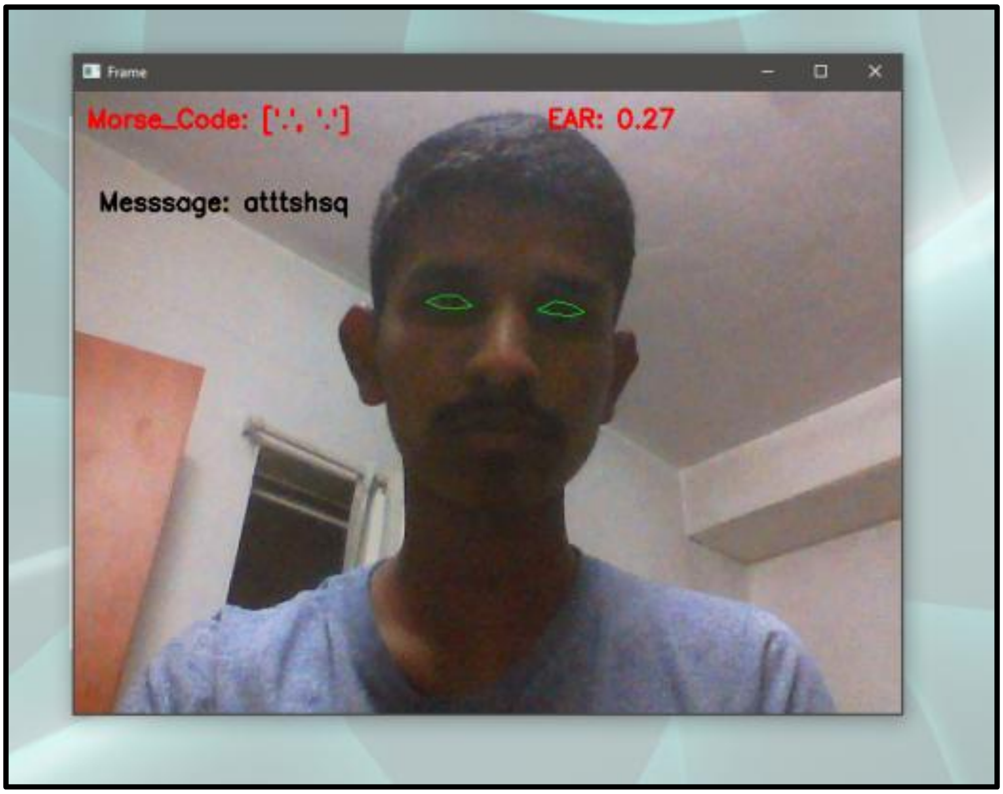

Figure 9: Input by user

Step 5: This is the history page of the user which consists of the list of words that user has used previously.

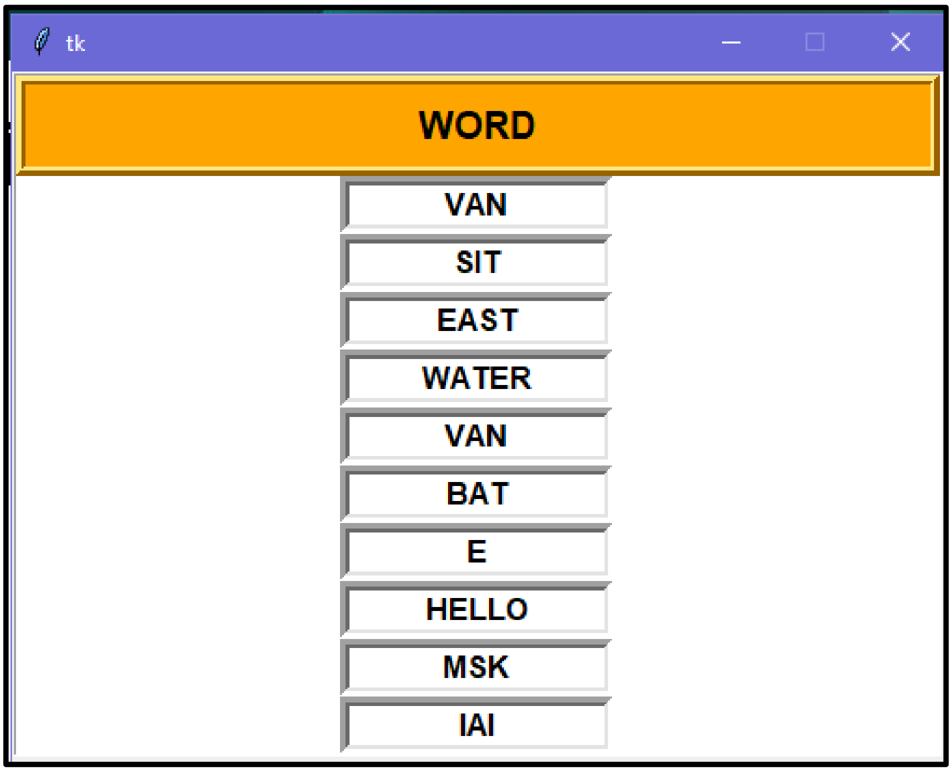

Figure 10: History

III. CONCLUSION

The world of technology is ever evolving and ever changing. The need for new technologies will never stop as people find new problems to solve or make new discoveries. People with disabilities face challenges every day, and developing new systems to aid them in fitting in is important. We proposed a facial gesture detection system for Morse code communications. An eye-blink detector and an open mouth detector are combined to permit communications via Morse code. A system will be developed for paralyzed people to communicate easily using human computer interaction based on morse code. In this we will develop a system for the paralyzed people for effortless communication. This system is cost effective.

REFERENCES

[1]. Tereza Soukupova and Jan ´ Čech -"Real-Time Eye Blink Detection using Facial Landmarks", EAR SVM algorithm approach ,21st Computer Vision Winter Workshop Luka Cehovin, Rok Mandeljc, Vitomir ˇ Struc (eds.) ˇ Rimske Toplice, Slovenia, February 3–5, 2016.

[2]. Dr. Kranthi Kumar , V. Sai Srikar , Y. Swapnika , V. Sai Sravani , N. Aditya-"A Novel Approach for Morse Code Detection from Eye Blinks and Decoding using OpenCV" OpenCV and dlib approach ,ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429 Volume 8 Issue V May 2020.

[3]. Guilei Hu, Yang Xiao , Zhiguo Cao , Lubin Meng, Zhiwen Fang , Joey Tianyi Zhou and Junsong Yuan-"Towards Real-Time Eye blink Detection in the Wild: Dataset, Theory and Practices" ,IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 15, 2020.

[4]. Link:- https://www.pyimagesearch.com/2017/04/24/eye-blink-detection-opencvpython-dl