Encrypted Communication using Face Detection and Eye Tracking using Morse Code

Submitted in partial fulfilment of the requirements

of the degree of

Bachelor of Engineering

by

Brandon Rodrigues	EU1162027
Manasvi Kotian	EU1162066
Nihar Vaidya	EU1162022

Under the guidance of

Ms Rashmi Bhat

Department of Computer Engineering

St. John College of Engineering and Management
University of Mumbai

2019-2020

CERTIFICATE

This is to certify that the project entitled **"Encrypted communication using face detection and eye tracking using Morse code"** is a bonafide work of

"Rodrigues Brandon" (EU1162027)

"Kotian Manasvi" (EU1162066)

"Vaidya Nihar" (EU1162022)

submitted to the University of Mumbai in partial fulfilment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"Computer Engineering"**.

Ms. Rashmi Bhat
Guide

Dr. Rahul Khokale
Head of Department

Dr.G.V. Mulgund
Principal

Project Report Approval for B. E.

This project synopsis entitled ***Encrypted communication using face detection and eye tracking using Morse code*** by ***Brandon Rodrigues, Manasvi Kotian***, ***Nihar Vaidya*** is approved for the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Examiners

1.----------

2.----------

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signature
Brandon Rodrigues (EU1162027)

Signature
Manasvi Kotian (EU1162066)

Signature
Nihar Vaidya (EU1162022)

Date:

Abstract

Digital Image Processing means processing digital image by use of computer algorithms, in order to get either an enhanced image or to extract some useful information. The main goal of the project is to not only create a platform for communication between individuals but also make it a secured one. This can be achieved by turning the platform for communication into a secured channel. The security is further improved by the utilization of a standardized code called as the Morse Code. Morse Code is a binary language which is created by a combination of '1' and '0'. The combinations represent words from any lexicon. The project is implemented when the user blinks his/her eyes to form a pattern or combination of the Morse Code. A normal blink refers to a '0' and a long blink refers to a '1'. If the user keeps his/her eyes open for a long period of time, it is to display the one alphabet is been completed. The system then uses various algorithms to decrypt this coded message and display it to the receiver. The system works such that the Morse Code is detected live from the user's live eye movements using a camera. The system finds its applications in military, wherein communication can be done in a safe environment using Morse code. It can be applied in hospitals for patients with physical disabilities and also video streaming. The unique language can be used to enhance the security parameter.

Table of Contents

List of Figures

Chapter 1

Introduction

Communication is the vital part of a person's daily life be it communication through speech, over an electronic device or through expressions. When a person communicates through an electronic device, the probability of leakage of data or loss of data is the highest. This project not only provides the user with a platform to communicate with each other but also do it in a secure manner. It aims to provide security by sending data in an encrypted format using a pre-existing code format and decrypts it for the intended user to read. The procedure is such that the user is able to select a method in which he/she chooses to communicate. The system detects the face and eyes of the user and counts the blinks and the pattern of the blink. Then synthesizing algorithms, the output is generated. One language used for communication is Morse code. Morse code is a combination of 1 and 0 in various pattern. Another language that has been introduced, unlike the Morse code, is not pre-determined.

1.1 Motivation

In today's day and age with cyber theft, cyber crime has been at such a risk. Having a save communication with a person is a difficult task. Many applications track the data that is been sent between the users. With the help of this application the users is able communicate with each other in a unique way. The system works on the human face detection and human eye blinks. As it's not a common practice to communicate in such a manner the probability of interception of the messages and malicious of them are minimized. Though videos of the users can be intercepted but the way of communication with the help of blinking of the eyes is unique. The same application is be used in high security conversations where a person is not allowed to speak, but needs to convey a message, the user may use the technique on communication with the help of Morse code and send the message. There are patients that are unable to communicate through speech or through actions, the proposed system would be very effective in such scenario [1]. Hospitals have now started to teach patients to communicate with the help of eye movement and blinking. Have an application capable to read each and every movement is much better to send the message to the doctor /or/close one. But the process to teach the patients Morse code would be a tedious task, to eliminate this scenario the scope of the project extends to include a new language which would be easy to understand and which would include shortcuts to certain words or phrases.

1.2 Problem Statement

The main goal for the development of this project is to provide the users of this application with an interface to communicate with each other [2]. The interface over which the users would communicate with each other would be a secure medium to share valuable information. In regular text messengers the text data may be intercepted by the attacker to gain the knowledge. With the help of sending data using Morse code language the leakage of data is be minimized and with proper implementation eliminated altogether. Apart from communicating with an individual this application helps in detecting faces among people and then decrypting the messages sent by them. The final objective of the project is to help doctors and their relatives to have a proper communication with patients who are unable to speak or communicate in a proper manner.

1.3 Objectives

The objectives are as follows:

- To successfully implement a system that takes input and provides ready cryptanalysis for the messages.
- To perform face detection and eye blinking monitoring procedure [3].

1.4 Scope

- To implement word guessing.
- To implement face recognition module, that would enhance the security aspect.
- To develop a new code language which would prove to be useful to patients, as the scope of learning would be reduced.

Chapter 2

Review of Literature

2.1 Morse code-based communication system focused on amyotrophic lateral sclerosis patients

This application is based on face recognition and facial features techniques to extract region of interest (which in this case are the eyes), by applying filters and cascade classifier with OpenCV implemented in Visual Studio. A software program was developed to transform blinking into words, along with a graphical user interface; the ALS patient has the possibility to structure sentences, thus, facilitating his/her communication process with relatives and medical staff [1]. ALS causes a loss of physical movement and the patients are unable to perform physical movements. This immobility makes it difficult for the patients to communicate with the relatives, nurses and doctors. To facilitate their communication, many applications are available in the market. But those applications are expensive hence the low-income patients are unable to afford it. Hence, this application was developed. It is named CodeBlink. CodeBlink was developed using Visual Studio C# .net and other modern technologies such as artificial intelligence that facilitate the interpretation and communication of these patients, because it incorporates default images and words which can be identified and selected quickly. The algorithm captures patient's face, eyes and blinking to achieve communication through the system by using Morse code the algorithm used in CodeBlink consists of five parts: acquisition, processing, coding, interpretation and visualization. Thus, the ALS patients can easily converse with the medical staff using this application. Fig. 2.1 describes the working of the given system.

The working of this system is very similar to the main proposed system, with the only drawback being, the scope of communication here is limited.

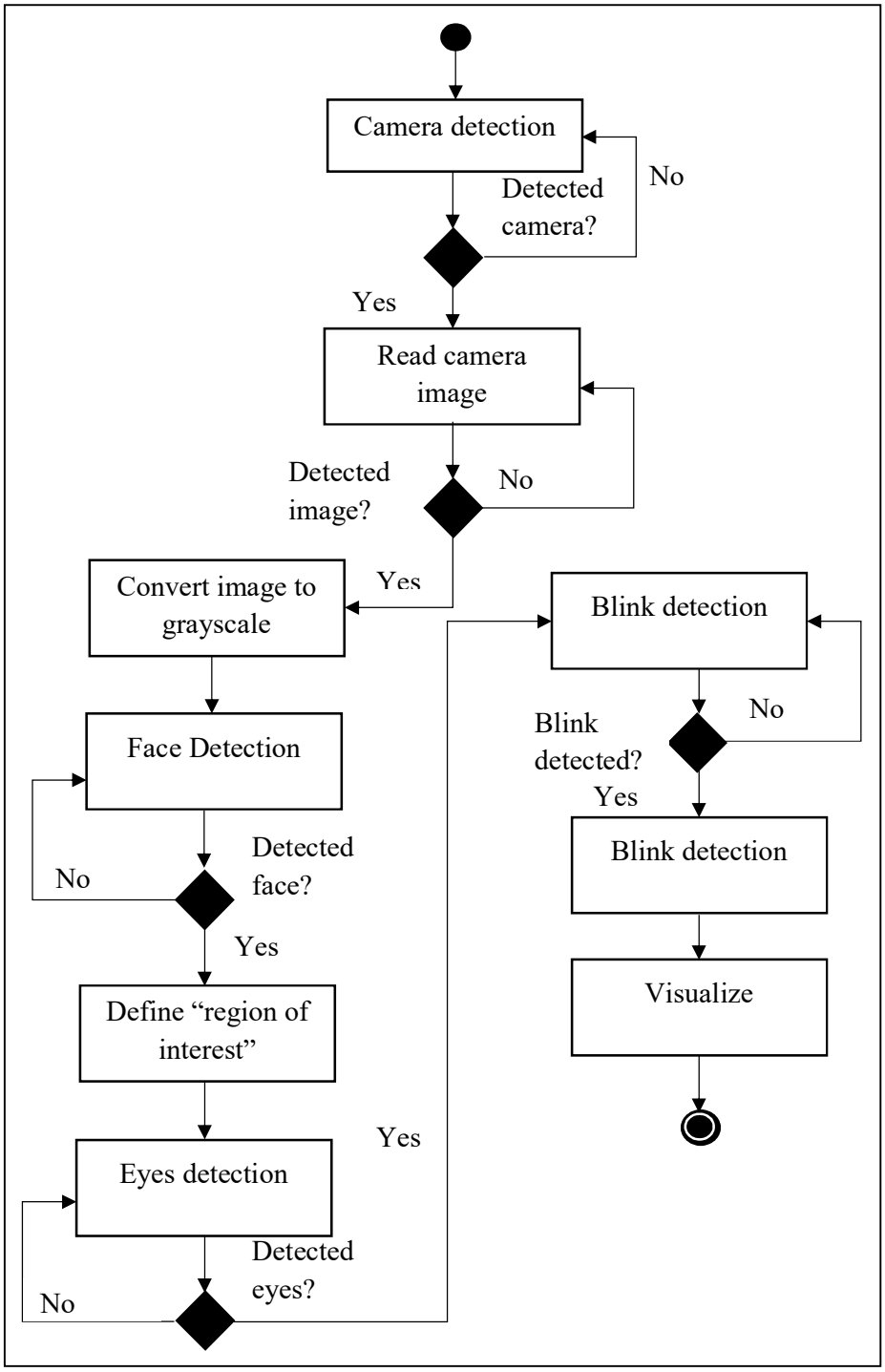

Fig. 2.1 Flowchart of Morse code-based communication system focused on amyotrophic lateral sclerosis patients

2.2 Human computer interaction by eye blinking on real time.

There are a lot of disabled people who can't control some parts of their bodies such as ALS and real time human – computer interaction systems can help them. A real time vision system is presented to provide a communication way to people has severe disabilities. Patients will be able to choose words on an alphabet tree which is designed on a binary tree by blinking right and left eye, thus they will make sentences [2]. Fig. 2.2 describes the working of the system. The scope of this flowchart is limited to the context displayed on the screen and nothing more, this is a limitation of this project.

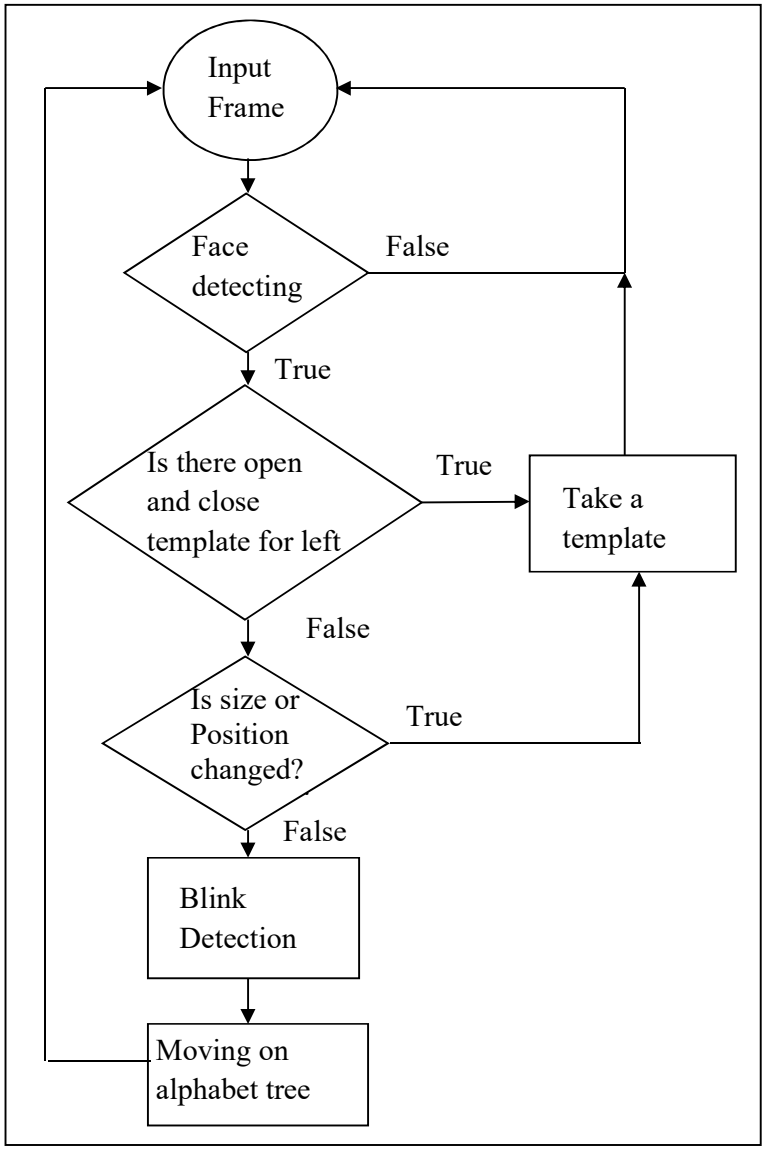

Fig. 2.2: Flowchart of Human computer interaction by eye blinking on real time.

2.3 Driver Fatigue Detection on Eye Tracking and Dynamic Template Matching.

A vision-based real-time driver fatigue detection system is proposed for driving safely. The driver's face is located, from color images captured in a car, by using the characteristic of skin colors. Then, edge detection is used to locate the regions of eyes. In addition to being used as templates for eye tracking in the next frame, the obtained eyes' images are also used for fatigue detection in order to generate some warning alarms for driving safety in the Fig. 2.3 shows the working of the system

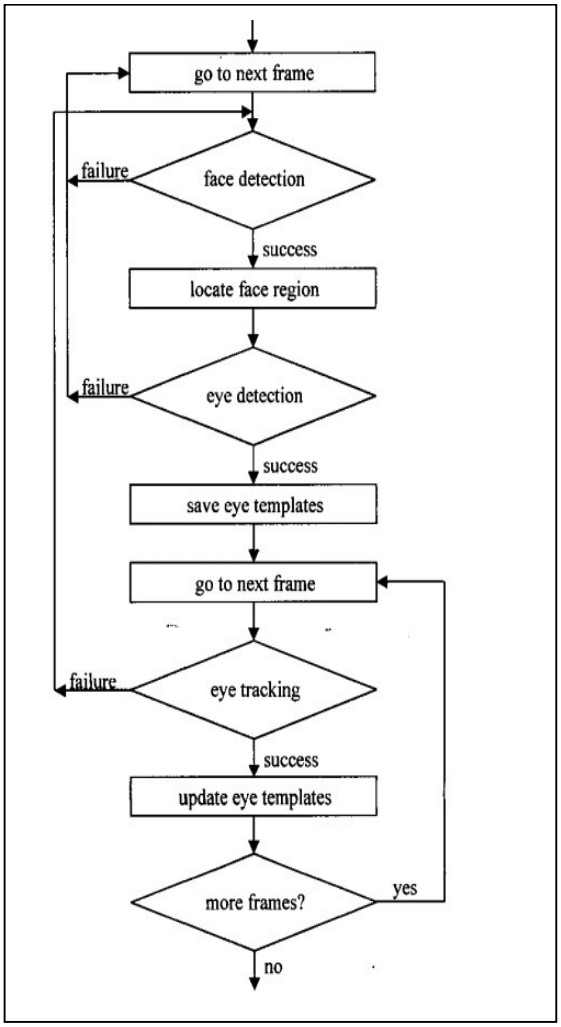

Fig. 2.3 Flowchart of Driver fatigue detection based on eye tracking and dynamic template matching [3].

Chapter 3

Requirement Analysis

This chapter gives the requirement for the project, requirement includes hardware, software, functional and non-functional.

3.1 Functional requirements

In the development of this project, some of the functional requirement could be:

- The system shall be able to take live input from the camera and focus on a singular face.
- The application shall also be able to play videos and detect faces and eyes.
- The system shall adhere to the new unique language and must be able to provide outputs accordingly in the most efficient manner.

3.2 Non-Functional requirements

In the development of this project, some of the non-functional requirement could be:

- The system shall be able to easily accessible to anyone with the basic understanding of the pre-assigned codes.
- The system shall run lag-free**.**

3.3 Minimum Hardware/Software requirements

3.3.1 Hardware Requirements

- Storage : 35 GB HDD storage
- RAM : 8 GB RAM
- Processor : i5 7^{th} gen and above
- Graphic card : GTX 750ti and above
- Camera : Inbuilt camera in laptop
  : External Webcam for desktop

3.3.2 Software Requirements

- Language use : Python 3.6 (should support dlib and tkinter libraries)
- Tools : OpenCV
  : DAT file(shape predictor)

Chapter 4

Design

This chapter specifies the structure of how a software system will be written and function, without actually writing the complete implementation. The design of the system is shown with the help of A. Data Flow Diagram (DFD) and B. Unified Modelling Language Diagram (UML). A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. UML diagram consists of Use Case Diagram, which is used to identify the events responsible for state changes, Sequence Diagram, which describes interactions among classes in terms of an exchange of messages over time and State Transition Diagram, which describes the behavior of a single object in response to a series of events in a system

4.1 Activity Diagram of Encrypted Communication using face detection and eye tracking using Morse code.

Activity diagram is an important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modelling the flow from one activity to another activity. The first method of using the system is by build in video.

The first method of using the system is by build in video. Fig. 4.1 explains the process of a single one-way eye capture based on the pre played video in the system. The system detects the face and the eyes of the user using the shape predictor file, the blinks performed by the user are been calculated and processed to display the final output.

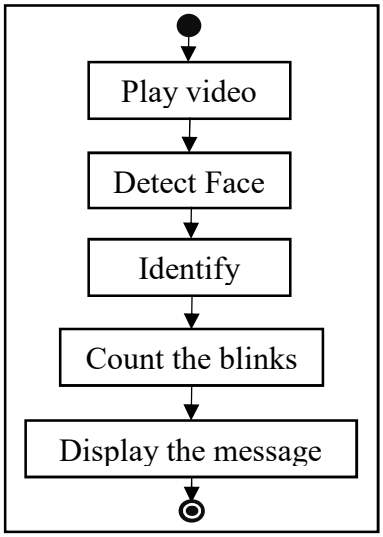

Fig. 4.1: Activity Diagram for Video as an input

The second application of the system is to work on the input received by the camera unit of the machine. Fig. 4.2 shows the process how the application detects and processes the blinks in a live camera feed [4].

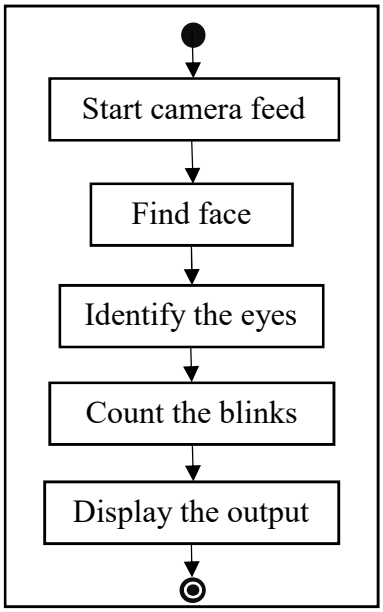

Fig. 4.2: Activity Diagram for Single person input

4.2 Activity Diagram of Encrypted Communication using face detection and eye tracking using Morse code.

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics. The working of the proposed system is been explained in Fig. 4.3.

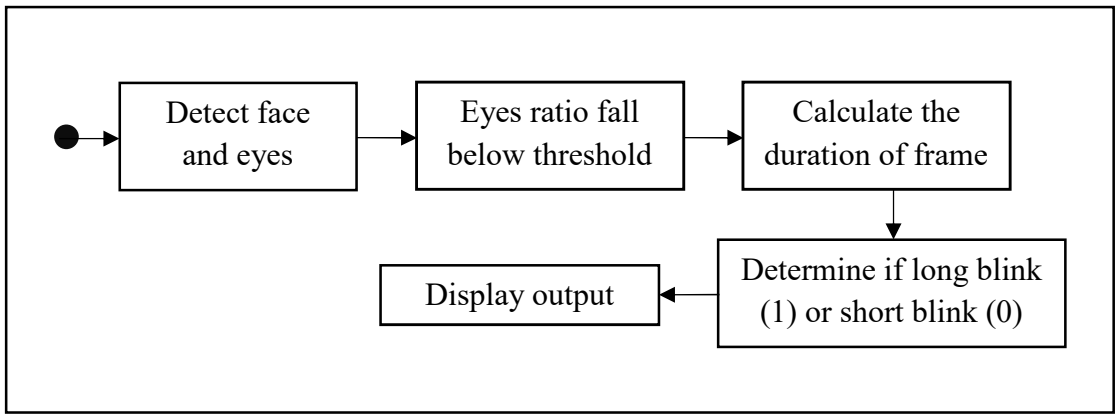

Fig. 4.3: Activity diagram of Encrypted communication using face detection and eye tracking using Morse code

4.3 Use Case Diagram of Encrypted Communication using face detection and eye tracking using Morse code.

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behaviour

Fig. 4.4 shows how a single user interacts with the application. It starts with the camera feed and detects eyes, refers with the Morse code and display the output. The process starts with the user and also ends with the same user thus it is a one-way communication.

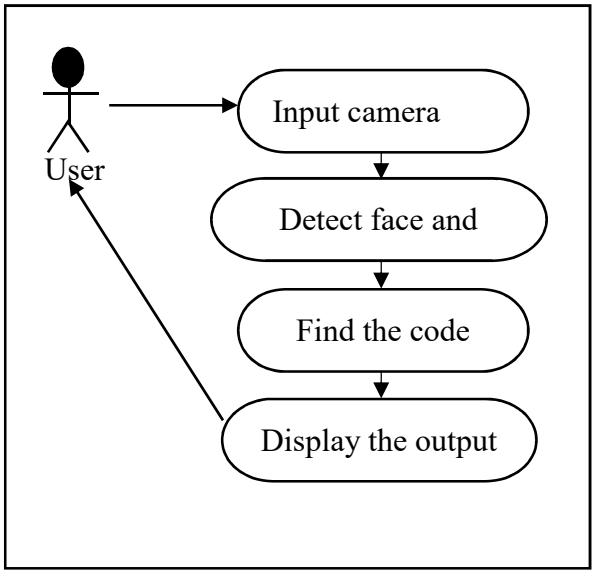

Fig. 4.4: Use case Diagram for single user/video input.

The working of the use case diagram is similar to all the functionality of the proposed system, hence no separate diagrams are needed to express the working of the system in video mode, Morse code mode and hospital mode.

4.4 Sequence Diagram of Encrypted Communication using face detection and eye tracking using Morse code.

Sequence diagrams are a popular dynamic modeling solution in UML because they specifically focus on lifelines, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends. Along with our UML diagramming tool, use this guide to learn everything there is to know about sequence diagrams in UML. The sequence in which the system access each aspect, which are the user, the application the CV module and the stored date, the process of this is been show in the following figure. The application follows the same procedure for all the working modes. The difference lie in the method

in which the input is been obtained or the database which is used to compare with the blinks. Fig. 4.5 shows the sequence in which the application performs its task in a single one-way process.

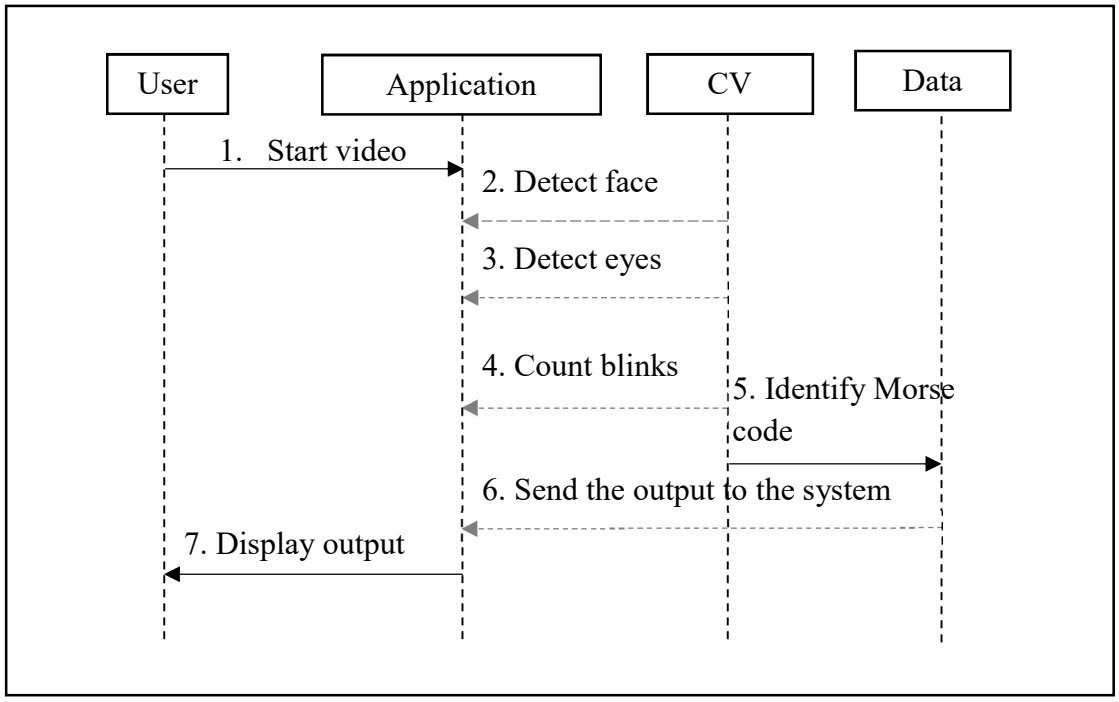

Fig. 4.5: Sequence Diagram for single user/video input.

The basic process in which the system performs its activities are similar for all the four functions. Any additional requirements necessary for the different modules would be explained in further. The changes only occur in the hospital language.

4.5 Data Flow Diagram of Encrypted Communication using face detection and eye tracking using Morse code.

Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports. A data-flow diagram has no control flow, there are no decision rules and no loops.

4.5.1 DFD Level 0 of Encrypted Communication using face detection and eye tracking using Morse code.

A level 0 data flow diagram (DFD), also known as a context diagram, shows a data system as a whole and emphasizes the way it interacts with external entities. Fig. 4.6 shows the simple application process of system in DFD level 0 format for the system. The user gains the desired output using application.

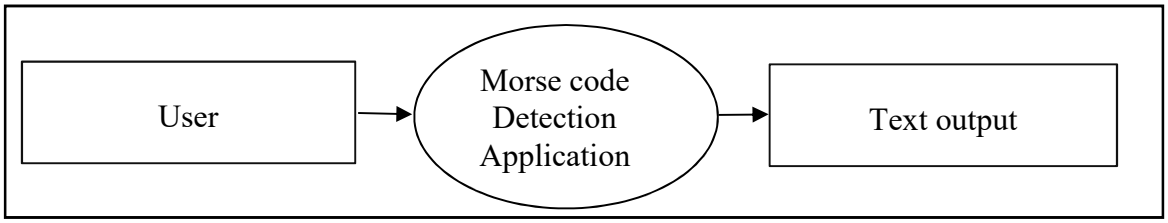

Fig. 4.6: DFD Level 0 diagram of Encrypted communication using face detection and eye tracking using Morse code

4.5.2 DFD Level 1 of Encrypted Communication using face detection and eye tracking using Morse code.

A level 1 DFD notates each of the main sub-processes that together form the complete system. User can think of a level 1 DFD as an "exploded view" of the context diagram. Fig. 4.7 shows the process of the system explained in DFD level 1. The application detects the face of the user and takes the blink action as the input and process with the data provided to refer i.e. Morse code and the output is printed in the text format.

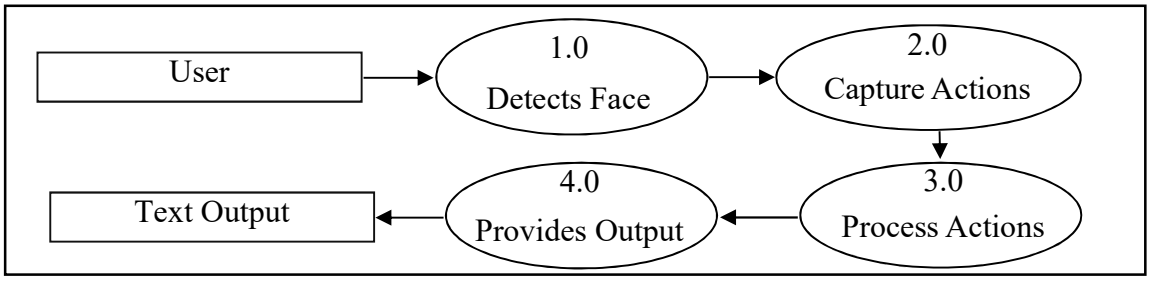

Fig. 4.7: DFD level 1 of Encrypted communication using face detection and eye tracking using Morse code.

4.5.3 DFD Level 2 of Encrypted Communication using face detection and eye tracking using Morse code.

A level 2 data flow diagram (DFD) offers a more detailed look at the processes that make up an information system than a level 1 DFD does. It can be used to plan or record the specific makeup of a system. To edit this DFD level 2 template, just sign up for a free Lucid chart account. User can then input the particulars of the system. Fig. 4.8 shows the process of the system explained in DFD level 2. Each expansion is explained in the detailed information. The further classifications are shown below.

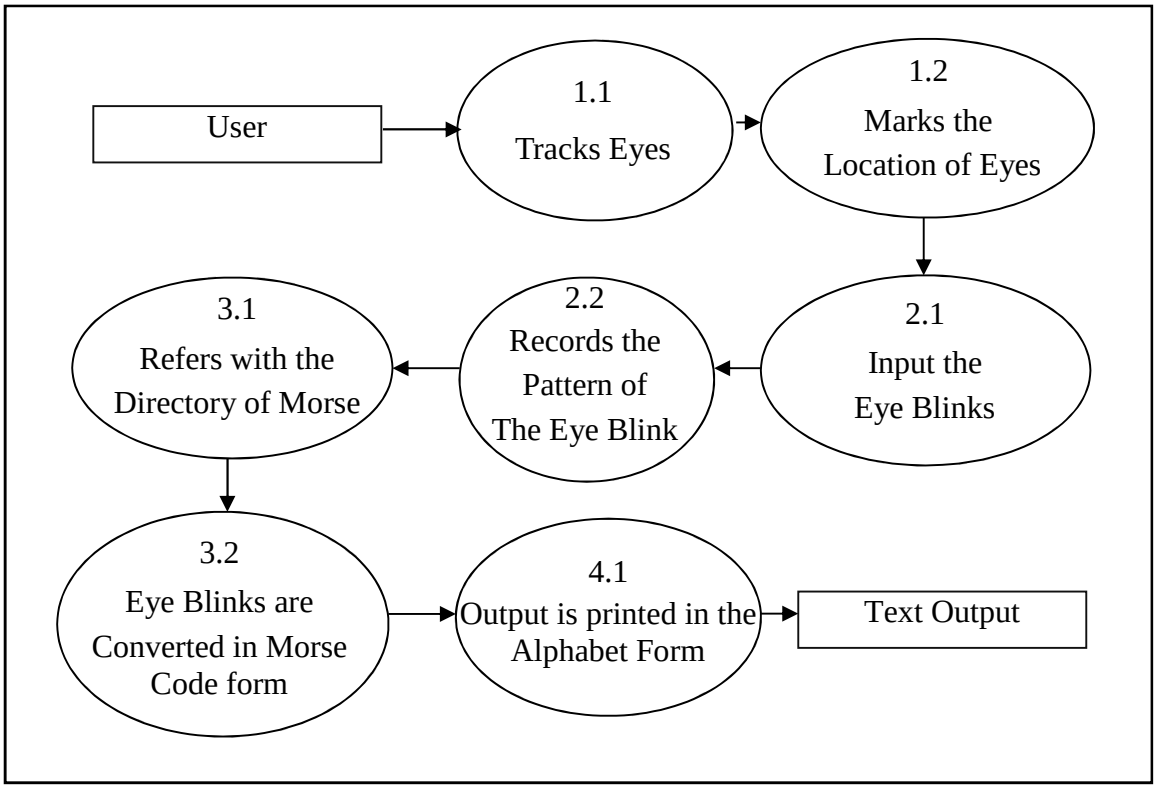

Fig. 4.8: DFD Level 2 of Encrypted communication using face detection and eye tracking using Morse code

Chapter 5

Report on Present Investigation

This chapter provides details regarding the complete process by the proposed system. All of the steps are explained in detail. This chapter describes the System Architecture followed by the system.

5.1 Methodology

The system generates an output using the predefined steps. Using dataflow diagrams as the reference, the system works in the following stages.

5.1.1 Selecting a model

The system provides the user to choose from four working models based on the requirement. The four models are: Morse code, Hospital language, Experimental unique language, and through video input. This dialogue box is the initial GUT component that is visible to the user as he executes the system.

5.1.2 Face detection

Once the user selects a module of their choice, the system tracks the face of the user. This is done by a fine called as “shape_predictor_68_face_landmarks.dat” [5], it is a python based file responsible for identifying facial features. It maps 64 co-ordinates on the face to identify it.

5.1.3 Eye tracking

The eyes are tracked based on the face co-ordinates. The motion of the eyes is tracked in frames. The eye aspect ratio, also called a as the EAR, is used to calculate the ratio of the eyes .The formula to calculate ear is:

$$Ear = \frac{(A + B)}{(2 \times C)}$$

where $A = (1, 5)$, $B = (2, 4)$, $C = (0, 3)$

where A, B and C are the co-ordinates of the eyes

The motion of the eyes is captured in frames. OpenCV is used to access the camera and capture frames [6]. As the user sends messages by blinking his eyes in a pattern, the eye blinks are considered as data [7]. The blinks are categorized into short blink, long blink and extremely long blink. Each blink represents a particular character as per the selected format. A blink is considered as a short blink if the eyes remain closed for 2 frames respectively. A blink is considered as a long blink if the eyes remain closed for consecutively 7 frames. An extremely long blink is when the eyes remain closed for consecutively 100 frames. The extremely long blink acts as a break and does not represent any character. Fig. 5.1 displays the 64 points describes in shape predictor module [8].

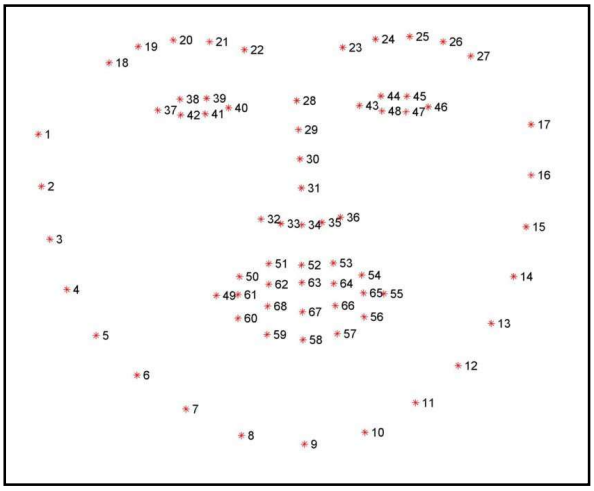

Fig. 5.1 Shape predictor – point marker [8]

Apart from shape predictor module, there are various other modules that are used to identify and detect the users face, but the complexity of using such modules varies.

5.2 Modules

The modules that are provided are Morse code, Hospital language, Experimental unique language, and video mode. These modules express the modes in which the proposed system is executed.

5.2.1 Morse code

A set of pattern denote alphabets in the English lexicon. The user blinks to form word and sentences of the Morse code. The Morse code is a binary language consisting of 0 and 1. The short blink represents zero and long blink represents one. To add a break between two characters we insert a break. To add a break between two words, the break must be applied twice. To exit the system, the user must press ‘q’. After the user has pressed ‘q’, the present dialogue box would be closed and another dialogue box would be opened which would display the message, given by the user, and options to quit the system or to re-launch it. Fig. 5.2 shows the Morse code language. Morse code is universally accepted as a mode of communication between army, as to communicate only two alphabets are required zero and one, and English alphabets are expressed as a combination of these two Morse code language that are zero and one

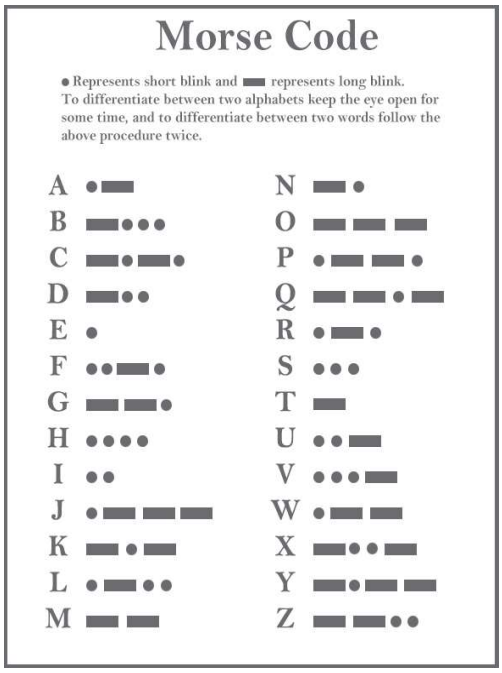

Fig. 5.2 Morse code

5.2.2 Hospital Language

This module is used by patients with physical disabilities who are unable to communicate verbally. In this module the combination of the short and long blinks denote most commonly used word by the patients. These words include “Yes”, “No”, “Help”, “Call the Doctor” etc. The difference in this module is that when the patient wants to exit the system, they have to keep their eyes shut for 3*100 frames as they are unable to use the keypad to exit. Fig. 5.3 shows the language created for patients.

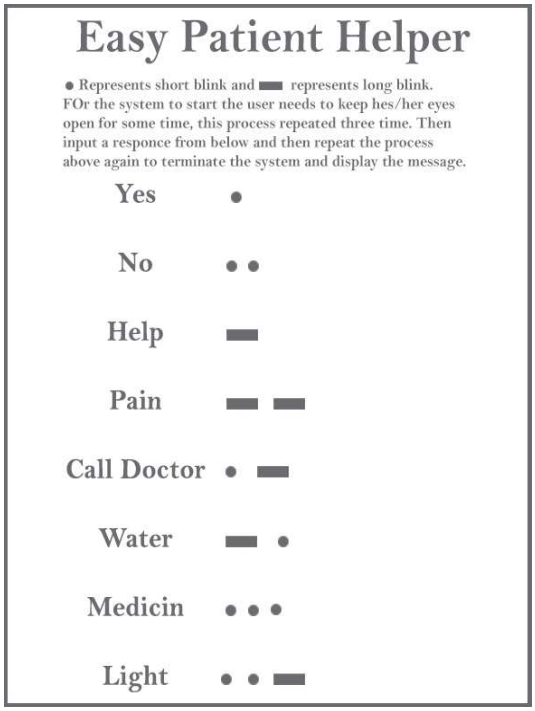

Fig. 5.3 Hospital code

5.2.3 Unique Language

Using a similar concept such as the Morse code, a unique language has been added to the system. Since Morse code is a standardized code, it has gained popularity and lost its security factor. This module used a combination of max three dash or dots, that in zero and one. At the experimental stage the application is able to accept only one word at a time. A single set of combination of zero and one is be assigned to two alphabets, this results in repetition, the application used probabilistic guessing of the final word, though the accuracy of this module would not be 100, further research is required.. Fig. 5.4 shows the proposed experimental

language. As the system has to use probabilistic analysis to determine the output word, the time required to run this module is higher than the rest. Enhancing the dictionary and spell checker algorithm would result less output delay.

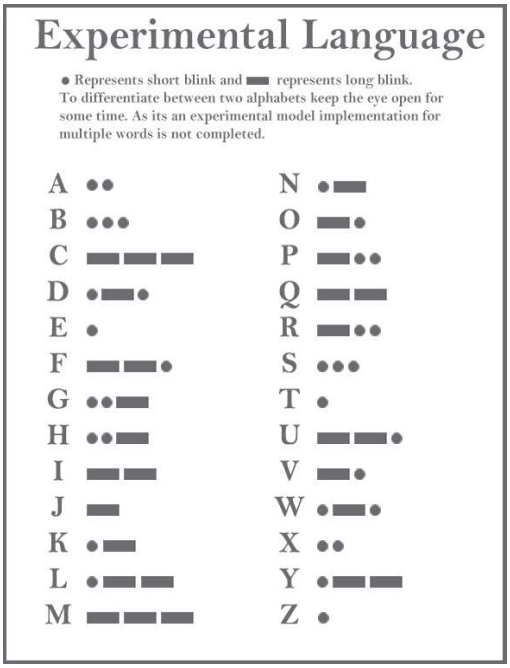

Fig. 5.4 Unique language

5.2.4 Video mode

This section takes video streaming as input. It detects the faces in a video it then captures frames using OpenCV and counts the number of frames to detect Morse code patterns [9]. Fig. 5.5 shows working of video mode.

Fig. 5.5 Video Streaming

5.3 Architecture Diagram

The application starts by the user starting the system, once the user has opened the application, the user is able to select between the modules presented above. The basic working of the system is been given in Fig. 5.6. The system gains access to the laptops/desktops webcam, after this process, the system along with 'shape_predictor_68_face_landmarks.dat' file, and this file allows the application to detect the eyes using 64 co-ordinate system. It stores the combination of blinks and then compares with the dictionary of the coded language. Once the output is been computed it display the output to the user.

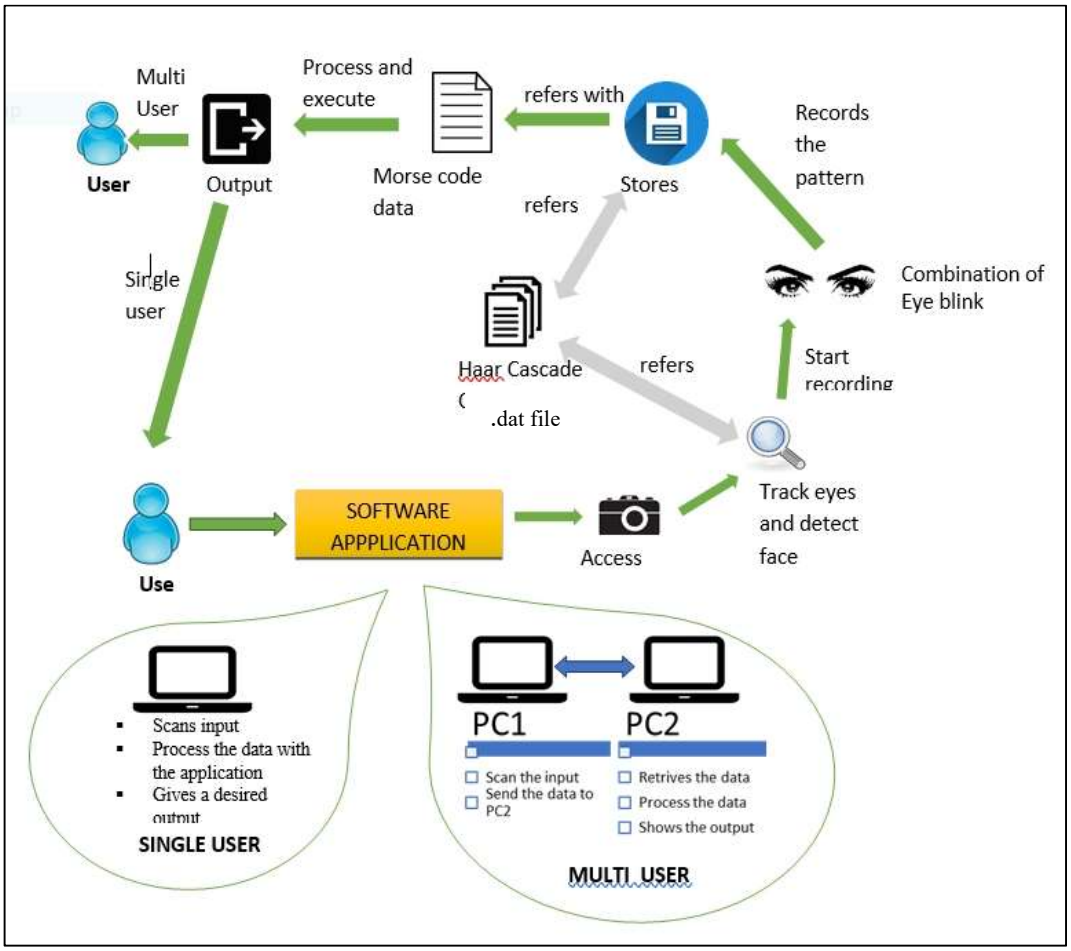

Fig. 5.6 Architecture Diagram

Chapter 6

Results and Discussions

After implementing the methodology, modules, algorithms and codes, the system will finally yield the expected outcome. The screenshots attached below will explain the procedure.

6.1 Results

With regard to the experimentation and the corresponding results, it is concluded that the system implementation is successful. System is able detect face using digital processing techniques and track the motion of the eyes and also provide automatic word prediction to accurately determine words and sentences. It can also be concluded that this method is effectively better than traditional text messaging as it includes disabled patients and caters to various requirements of the user.

6.2 Discussion

Users is able to select any one of the following options displayed in Fig. 6.1. Each individual section has further options of either viewing the code or running it to communicate. The options available for the user are the modules that are discussed in the previous chapters, after selecting the modules the proposed system uses the shape predictor file to display the outputs. The working of face recognition is been implemented in the Morse code section. With the implementation of face recognition the security of the proposed system highly increases in application such as for military use, encrypted communication by adding a new line of safety.

Modified code uses an experimental language developed for the system which is uses a new combination of one and zero, and used probabilistic analysis. The Hospital module runs continuously, but only starts detecting eyes once three ‘B’ have been detected and terminated at the same [10].

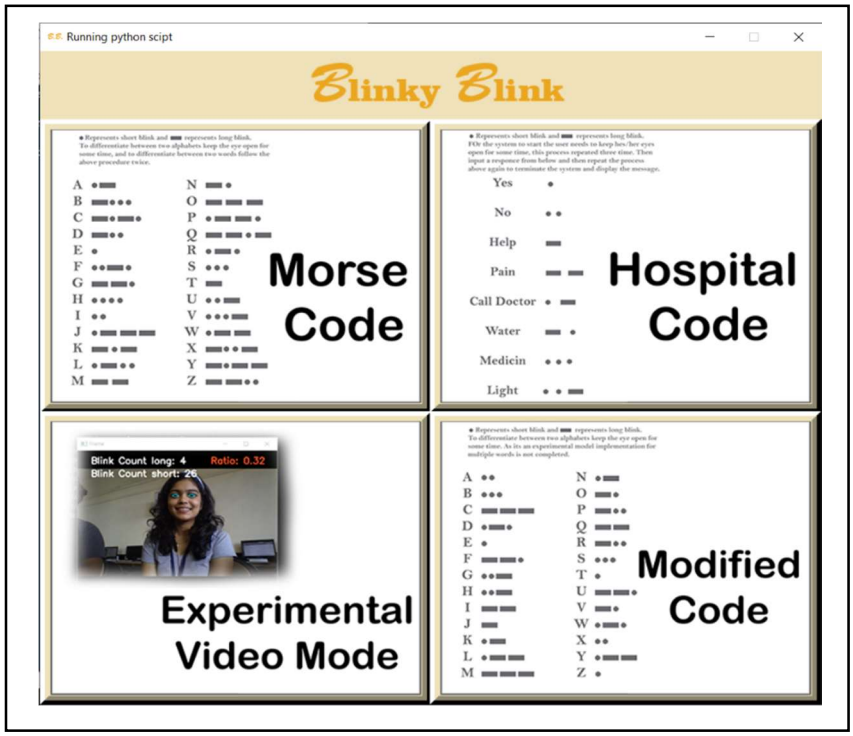

Fig. 6.1: User Interface

Fig. 6.2 displays the output of the text message after successfully running the program. The above implementation was conducted in the Hospital module, where the patient blinked to send the “Help Me” message.

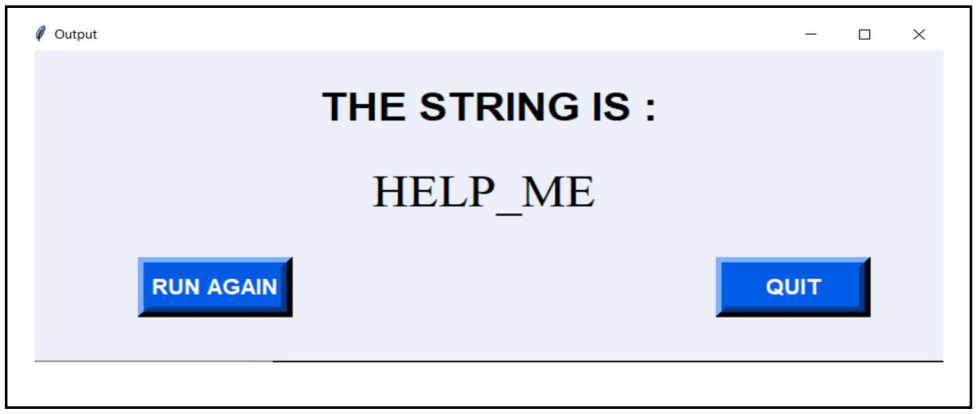

Fig. 6.2: Displayed message

Chapter 7

Conclusion

The Encrypted Communication using Face detection and Eye Tracking using Morse Code has been successfully implemented. Users can communicate on this platform by choosing any of the four modules: Morse, Hospital, Video and Unique. Upon selecting the desired module, the user can successfully send and receive messages by adhering to the code format, by blinking eyes accordingly. The system detects the face and tracks the motion of the eyes to extract the message and successfully encrypts and decrypts it. It uses Python and its libraries to conduct face detection and grammar libraries to predict plausible responses. The system accurately captures frames, both in live streaming as well as for video files.

References

[1] Rosado, Franklin, et al. "Morse code-based communication system focused on amyotrophic lateral sclerosis patients." *2016 XXI Symposium on Signal Processing, Images and Artificial Vision (STSIVA)*. IEEE, 2016.

[2] Aksu, Doğukan, and M. Ali Aydın. "Human computer interaction by eye blinking on real time." *2017 9th International Conference on Computational Intelligence and Communication Networks (CICN)*. IEEE, 2017.

[3] Horng, Wen-Bing, et al. "Driver fatigue detection based on eye tracking and dynamic template matching." *IEEE International Conference on Networking, Sensing and Control, 2004*. Vol. 1. IEEE, 2004.

[4] Danisman, Taner, et al. "Drowsy driver detection system using eye blink patterns." *2010 International Conference on Machine and Web Intelligence*. IEEE, 2010.

[5] Adrian Rosebrock, "*Eye blink detection with OpenCV and Python Demo*", 2017. Youtube, 13 April 2017, Available: https://www.youtube.com/watch? v=0w8FaiKP5h8 .

[6] Adrian Rosebrock, "*Python Programming Tutorial python programming*", 2017, Available: https://pythonprogramming.net/haar-cascade-face-eye-detection-python-opencv/ .

[7] Eriksson, Martin, and Nikolaos P. Papanikotopoulos. "Eye-tracking for detection of driver fatigue." *Proceedings of Conference on Intelligent Transportation Systems*. IEEE, 1997.

[8] Adrian Rosebrock, "*Facial landmarks with dlib, OpenCV, and Python*", 2019, Available: https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/ .

[9] Avinab Saha, Read, "*Write and Display a video using OpenCV (C++/ Python)*", 2017, Available: https://www.learnopencv.com/read-write-and-display-a-video-using-opencv-cpp-python/ .

[10] Adrian Rosebrock, **"***OpenCV with Tkinter tutorials for GUI based design***"**, 2016**,** Available: https://www.pyimagesearch.com/2016/05/23/opencv-with-tkinter/ **.**

Technologies Used

1. Python:

Python is an interpreter, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace.

2. OpenCV:

OpenCV-Python. OpenCV-Python is a library of Python bindings designed to solve computer vision problems. ... OpenCV-Python makes use of Numpy, which is a highly optimized library for numerical operations with a MATLAB-style syntax. All the OpenCVarray structures are converted to and from Numpy arrays.

3. Dlib:

Dlib is a general purpose cross-platform software library written in the programming language C++. ... It is open-source software released under a Boost Software License. Since development began in 2002, Dlib has grown to include a wide variety of tools.

4. Imutils:

Imutils are a series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV and both Python 2.7 and Python 3.

5. Tkinter:

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard Linux, Microsoft Windows and Mac OS X installs of Python. The name Tkinter comes from Tk interface.

Publication

Sr.No.	Paper title	Author	Publication details
01	Encrypted Communication using Face Detection and Eye Tracking using Morse code	Brandon Rodrigues Manasvi Kotian Nihar Vaidya Prof. Rashmi Bhat	St. John Journal of Engineering Science

Acknowledgement

We have great please in presenting the report on **Encrypted Communication using Face Detection and Eye Tracking using Morse code**. We take the opportunity to express my deep regards towards the ones who offered their valuable guidance in the hour of need. Prominent among them are our guide **Ms. Rashmi Bhat** and our Head of the Department **Dr. Rahul Khokale** . The ideas suggested by them are incorporated in this project topic report. Thank you for supporting us and for making our research possible.

We express our warm thanks to the Principal, **De. G. V. Mulgund** for their support and guidance. Furthermore, we would also like to express our special thanks to our friends for the interesting and nice time we had when doing Graduation together.

Last but not the least we would like to thank our family, especially our parents for always putting our education first and supporting us in all stages of my life.

Brandon Rodrigues(EU1162027)

Manasvi Kotian(EU1162066)

Nihar Vaidya(EU1162022)