EYE / FACIAL GESTURE SYSTEM CONTROLLER BASED ON AI / ML

Project Report

Submitted in Partial Fulfillment of the Requirements for the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

By

NANDINI (1900950100046)

SHIV PRATAP SINGH SENGAR(1900950100073)

SHRISHTY SHAKYA(1900950100075)

SWATI RAJ (1900950100080)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MGM's College of Engineering & Technology, Noida

May, 2023

TABLE OF CONTENTS

DECLARATION

I hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature:
Name: Nandini
Roll No.: 1900950100046

Signature:
Name: Shiv Pratap Singh Sengar
Roll No.: 1900950100073

Signature:
Name: Shrishty Shakya
Roll No.: 1900950100075

Signature:
Name: Swati Raj
Roll No.: 1900950100080

Date: 05.05.2023

CERTIFICATE

This is to certify that Project Report entitled "EYE / FACIAL GESTURE SYSTEM CONTROLLER BASED ON AI/ML" which is submitted by team Nandini, Shiv Pratap Singh Sengar, Shrishty Shakya, Swati Raj in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science and Engineering of MGM's College of Engineering and Technology which is affiliated by AKTU Lucknow, is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Date: 05.05.2023

Supervisor Signature:

Name of Supervisor: Mrs. Karamjeet Kaur

Designation: Head Of Department

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Mrs. Karamjeet Kaur Department of Computer Science & Engineering, MGM CoET, Noida for his/her constant support and guidance throughout the course of our work. His/her sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his/her cognizant efforts that our endeavour have seen light of the day. We also take the opportunity to acknowledge the contribution of Mrs. Karamjeet Kaur Head, Department of Computer Science & Engineering, MGM CoET, Noida for his/her full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature:
Name: Nandini
Roll No.: 1900950100046

Signature:
Name: Shiv Pratap Singh Sengar
Roll No.: 1900950100073

Signature:
Name: Shrishty Shakya
Roll No.: 1900950100075

Signature:
Name: Swati Raj
Roll No.: 1900950100080

Date: 05.05.2023

ABSTRACT

We have presented a novel method of human-computer interaction in this technology that is based on eye and facial gestures. It is software that uses an image sensing device to communicate with gestures. In order to synthesise natural gestures, acts gesture and symbol gesture were compared. Using Data-Driven Gesture Synthesis: Evaluating Gesture Units Hypoglossal facial anastomosis and jump interposition graft in the treatment of facial reanimation: Evolution The initial goal of the software is ,in the first time to detect user's face and their eyes in the second time. The eyes gesture recognition is based on fuzzy inference system. The fuzzy inference system is the foundation for the eyes gesture recognition. Eye tracking devices have historically fallen into two categories. The first is passive, and focused on detecting the gaze of the user relative to the rest of the world and in particular what elements of the visible field are currently being focused upon. The second is more active, and considers the eye not as simply a means of observation, but a means of control as well. We think that a device capable of identifying deliberate movements of the eye area (pupils, eyelids and eyebrows), we can provide a new means of interaction that could replace or complement more standard interfaces.

The human eyes, a beautiful and interactive organ in the human body, have unique physical, photometric, and motion characteristics. These characteristics provide vital information required for eye detection and tracking. In our daily lives, a person's emotional state, mental occupancy, and needs can be judged by the person's eyes movements. Through our eyes, we identify the properties of the visual world and collect the information essential to our lives. Moreover, in the field of image and video processing, eyes play a vital role in the process of face detection and recognition. Gesture recognition provides real-time data to a computer to make it fulfill the user's commands. Motion sensors in a device can track and interpret gestures, using them as the primary source of data input

Data Flow Diagrams: Function modelling is represented with the help of DFDs. DFD is the graphically representation of data. It shows the input, output and processing of the system. When we are trying to create our own business, website, system, project then there

is need to find out how information passes from one process to another so all are done by DFD. The pupil and iris being darker than their surroundings are commonly considered reliable features for eye detection. We can use a skin-color model and introduce an algorithm to locate the pupils by searching for two dark areas that fulfill specific anthropometric requirements. Their technique, however, cannot perform well in direct light conditions due to limitation of the skin-color model. The typical eye structure used in gaze tracking applications is demonstrated in Figure. The modeling of gaze direction is based either on the visual axis or on the optical axis. The visual axis, which forms the line of sight (LoS) and is considered the actual direction of gaze, is the line connecting the center of the cornea and the fovea. The optical axis, or the line of gaze (LoG), is the line passing through the centers of pupil, cornea, and the eyeball. The center of cornea is known as the nodal point of the eye. System integration is the process of connecting different sub-system(components) or modules into a single larger system or program that function as one. With regards to software solution, system integration is typically defined as the process of linking together various modules. Risk Identification forms an important part in Software Development Life Cycle. This gives a view of all the possible risks to the system and whether they can be controlled or not.

List Of Figures

CHAPTER 1
INTRODUCTION AND BACKGROUND OF PROJECT

INTRODUCTION

The human eyes, a beautiful and interactive organ in the human body, have unique physical, photometric, and motion characteristics. These characteristics provide vital information required for eye detection and tracking. In our daily lives, a person's emotional state, mental occupancy, and needs can be judged by the person's eyes movements. Through our eyes, we identify the properties of the visual world and collect the information essential to our lives. Moreover, in the field of image and video processing, eyes play a vital role in the process of face detection and recognition. The notion that useful conclusions could be drawn from information about the movement of a person's eyes has been around for well over a century. For a sighted person, the primary means of interacting with the world is through visual input, so it is natural to assume that information about what a person is looking at in any given moment would be instrumental in determining how that person is interacting with the world. In the field of human computer interface design, knowledge about such interactions can be critical to designing a powerful, intuitive and ultimately helpful user interface.

1.1. LITERATURE REVIEW

A literature review establishes familiarity with and understanding of current research in a particular field before carrying out a new investigation. Conducting a literature review should enable you to find out what research has already been done and identify what is unknown within your topic. The literature was studied to address the aims, understanding of the research area, focus on the research questions, planning of the data collection approach, clarification of the meaning of the terms and proper identification of the framework. The most important task was to understand the research domain in which eyes detection and cursor movement of a mouse is involved.

The eye gesture control system directly interacts with the vision of the human eyes and then controls the system. Eye gesture, a real-time gesture assurance programming which controls a computer mouse cursor by using the user's eye gestures. Only requirement to operate the mouse system is, individuals having at least one eye with good vision and ability to control the computer. Its users can be grown-ups and youngsters with cerebral paralysis, spinal rope wounds, mental wounds, ALS, different sclerosis, Brainstem strokes, and so on. Eye gesture control system can be utilized as a part of homes, workplaces, schools, healing centers, and long-haul mind offices. By looking at the control of a system that is display on a screen, a person can run a computer software, operate a computer mouse, and access the internet and also email.

Provide a cheap eye-tracking system, to control the cursor of a computer with eyes. Allow physically disabled people to use computers. To control a computer and communicate with other systems. To provide a real-time accurate eyes gesture control system. To provide a hand free mouse control system. To provide a complete generic eye-gesture mouse control system. To provide a complete wire free mouse control system. Easy to control cursor movement of a mouse. Going with the literature, we are focusing on customer satisfaction who are physically disabled. In 2018 , an eye tracking algorithm based on Hough transform was developed. This system detects the face and eyes of a person. It uses a webcam to detect user's face and eyes. The issue in this system is of real-time tracking and time speed issue.The system is quite slow and it needs a high-quality computer system to work properly which is costly.Eye tracking began in the late 1800s with mechanical devices that tracked light reflection patterns or even materials directly embedded in the cornea. With the growth of photography and video recording technology, far more reliable and less invasive means were developed to simply observe a user's eye motions during long periods of activity. These recordings would then be analyzed manually, often on a painstaking frame-by-frame basis, generating mountains of data to analyze. The task was daunting and even in a perfectly performed experiment, the data could very easily defy all attempts at understanding. The growth of computing technology eventually eased the data analysis task to the point of feasibility, and eye-tracking experiments became more popular, and devices for incorporating eye gaze data into psychological and medial studies began to

appear. It is only recently, however, that computing power and video recording capability have become inexpensive and powerful enough to cross the gap into the demanding, realtime world of interface design. It is this frontier of the eye-tracking field that our project seeks to explore.

According to M. Betke, J. Gips and P. Fleming. "The camera mouse: Visual tracking of body features to provide computer access for people with severe disabliites." IEEE Transactions on Neural Systems and Rehabilitation Engineering, 10:1, pages 1-10, March 2002. The "Camera Mouse" system has been developed to provide computer access for people with severe disabilities. The system tracks the computer user's movements with a video camera and translates them into the movements of the mouse pointer on the screen. Body features such as the tip of the user's nose or finger can be tracked. The visual tracking algorithm is based on cropping an online template of the tracked feature from the current image frame and testing where this template correlates in the subsequent frame[1].

A study entitled R.J.K. Jacob and K.S. Karn. “Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises (Section Commentary),” in The Mind's Eye:Cognitive and Applied Aspects of Eye Movement Research, ed. by J. Hyona,R. Radach, and H. Deubel, pp. 573-605, Amsterdam, Elsevier Science (2003). For usability analysis, the user's eye movements are recorded during system use and later analyzed retrospectively; however, the eye movements do not affect the interface in real time. As a direct control medium, the eye movements are obtained and used in real time as an input to the user–computer dialogue[2].

According to C. Morimoto and M. Mimica. "Eye gaze tracking techniques for interactive applications."Computer Vision and Image Understanding 98", pages 4-24, 2005. Eye gaze data have been used in ophthalmology, neurology, psychology, and related areas to study oculomotor characteristics and abnormalities, and their relation to cognition and mental states. There are more recent applications for research in marketing and advertising, as well as in human factors engineering to evaluate computer interfaces and web sites, but they are still confined to controlled environments[3].

According to Tom M. Mitchell, ―Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.Online data sets and implementations of several algorithms are available on a Web site. No prior background in artificial intelligence or statistics is assumed. For advanced undergraduates and graduate students in computer science, engineering, statistics, and social sciences, as well as software professionals[4].

A study entitled Ethem Alpaydin, ―Introduction to Machine Learning (Adaptive Computation and Machine Learning), The MIT Press 2004. he researchers in these various areas have also produced several different theoretical frameworks for understanding these methods, such as computational learning theory, Bayesian learning theory, classical statistical theory, minimum description length theory, and statistical mechanics approaches. These theories provide insight into experimental results and help to guide the development of improved learning algorithms. A goal of the series is to promote the unification of the many diverse strands of machine learning research and to foster high quality research and innovative applications. This series will publish works of the highest quality that advance the understanding and practical application of machine learning and adaptive computation[5].

A study entitled S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach‖, Prentice Hall, Third Edition, 2009. Artificial Intelligence (AI) is a big field, and this is a big book. We have tried to explore the full breadth of the field, which encompasses logic, probability, and continuous mathematics; perception, reasoning, learning, and action; and everything from microelectronic devices to robotic planetary explorers. The book is also big because we go into some depth. The subtitle of this book is "A Modern Approach." The intended meaning of this rather empty phrase is that we have tried to synthesize what is now known into a common framework, rather than trying to explain each subfield of AI in its own historical context. We apologize to those whose subfields are, as a result, less recognizable[6].

1.2. PROBLEM DEFINITION

Gesture recognition provides real-time data to a computer to make it fulfill the user's commands. Motion sensors in a device can track and interpret gestures, using them as the primary source of data input. A majority of gesture recognition solutions feature a combination of 3D depth-sensing cameras and infrared cameras together with machine learning systems.Machine learning algorithms are trained based on labeled depth images of hands, allowing them to recognize hand and finger positions. We wish to present a novel method of human-computer interaction in this technology that is based on eye and facial gestures. It is software that uses an image sensing device to communicate with gestures. In order to synthesise natural gestures, acts gesture and symbol gesture were compared. Using Data-Driven Gesture Synthesis: Evaluating Gesture Units Hypoglossal facial anastomosis and jump interposition graft in the treatment of facial reanimation: Evolution The initial goal of the software is ,in the first time to detect user's face and their eyes in the second time. The eyes gesture recognition is based on fuzzy inference system. The fuzzy inference system is the foundation for the eyes gesture recognition.

Gesture recognition consists of three basic levels:

1. Detection: With the help of a camera, a device detects hand or body movements, and a machine learning algorithm segments the image to find hand edges and positions.
2. Tracking: A device monitors movements frame by frame to capture every movement and provide accurate input for data analysis.
3. Recognition: The system tries to find patterns based on the gathered data. When the system finds a match and interprets a gesture, it performs the action associated with this gesture. Feature extraction and classification in the scheme below implements the recognition functionality.
4. Keep in mind that hand tracking and gesture recognition are not the same things. Both technologies are supposed to use hands for human-machine interaction (HMI) without touching, switching, or employing controllers. Sometimes, systems for

hand tracking and gesture recognition require the use of markers, gloves, or sensors, but the ideal system requires nothin but a human hand.

5. Why may people want to use gestures instead of just touching or tapping a device? A desire for contactless sensing and hygiene concerns are the top drivers of demand for touchless technology.
6. Gesture recognition can also provide better ergonomics for consumer devices. Another market driver is the rise of biometric systems in many areas of people's lives, from cars to homes to shops.
7. During the coronavirus pandemic, it's not surprising that people are reluctant to use touchscreens in public places. Moreover, for drivers, tapping a screen can be dangerous, as it distracts them from the road.

In other cases, tapping small icons or accidentally clicking on the wrong field increases frustration and makes people look for a better customer experience.Real-time hand gesture recognition for computer interactions is just the next step in technological evolution, and it's ideally suited for today's consumer landscape.Besides using gestures when you cannot conveniently touch equipment, hand tracking can be applied in augmented and virtual reality environments, sign language recognition, gaming, and other use casesThe high cost of touchless sensing products is one of the major challenges of this technology, along with the complexity of software development for HGR.To create a robust system that detects hand positions, a hand tracking solution requires the implementation of advanced machine learning and deep learning algorithms, among other things.

1.3. BRIEF INTRODUCTION OF THE PROJECT

Vision is probably the most important sense for human beings. As a consequence, our way of behaviour and thinking is also often based on visual information.When trying to perform complex information especially in situations where humans are involved, it is of great benefit if some information can be obtained from images. This is the field of image processing and computer vision. OpenCV is the library used in this project. It can also be used in Python programming language. Simple and complex image processing algorithms are available in the library. Face detection is one of them and face detection can be executed

within python with OpenCV library.Now-a-days many people are getting disabled with hands and any other disabilities by which a candidate cannot be able to use keyboard and mouse. So, for disabled persons we had a thought on vision which can be responsible for controlling a computer using mouse and keyboard.The eye gesture control system directly interacts with the vision of the human eyes and then controls the system.Eye gesture, a real-time gesture assurance programming which controls a computer mouse cursor by using the user's eye gestures.Only requirement to operate the imouse system is, individuals having at least one eye with good vision and ability to control the computer.Its users can be grown-ups and youngsters with cerebral paralysis, spinal rope wounds, mental wounds, ALS, different sclerosis, Brainstem strokes, and so on.Eye gesture control system can be utilized as a part of homes, workplaces, schools, healing centers, and long-haul mind offices. By looking at the control of a system that is display on a screen, a person can run a computer software,operate a computer mouse, and access the internet and also email.Provide a cheap eye-tracking system, to control the cursor of a computer with eyes. Allow physically disabled people to use computers.

1.4. Objectives

1. To control a computer and communicate with other systems. To provide a real-time accurate eyes gesture control system
2. To provide a hand free mouse control system. To provide a complete generic eye-gesture mouse control system
3. To provide a complete wire free mouse control system. Easy to control cursor movement of a mouse.

1.4.1. Feature Tracking

The tracking system is the first step in the eye-tracking device and is the most directly in contact with the user.Tracking processes all of the data from the camera inputs, presents the calibration and parameter-tweaking interfaces to the user, and performs computer vision-related algorithms to determine the location of the user's pupils and eyebrows. From a user-interface perspective, the operation of the device is as follows:

1. When the program begins, the user is prompted to position the camera and headgear properly so that the full regions of both eyes are clearly visible and unobstructed in the video.
2. Then the user is prompted to select the appropriate eye regions from a video snapshot.

   The selection of the eye region is very important because it allows us to significantly narrow our search space, ensure that only eye-related features appear on the image, and cut down on external interference with the tracking process.
3. With the regions selected, detailed tracking windows appear and the user is prompted to configure (via a set of trackbars) the parameters that determine tracking. This is certainly the clunkiest part of the system, and the part that would benefit greatly from improvement in a more polished version of the device.Specifically, the trackbars control the thresholding level applied to the input images.
4. The user does not need to know any technical details since the best-guess estimate of pupil and eyebrow position are superimposed on the camera input, and the user need only slide the various trackbars until the estimates are steady and follow his or her movements.
5. After the parameters are set, calibration mode begins. In this stage, the user is prompted conduct a few eye and eyebrow movements while the program trains.
6. The specific required actions are looking left, right, up and down, and raising the eyebrows. Once calibration is complete, the user is ready to begin using the device.
7. The device can be recalibrated or disabled at any time with a simple keystroke

The tracking algorithms are both based heavily on the contrasts and contours contained in the image of the eye. The pupil recognition algorithm is borrowed from Zafer Savas' TrackEye software. The image is first thresholded and then run through a Canny edge detector. Then contours are found and the contours are filtered to extract the largest, most circular closed contour available. In a sufficiently controlled image with proper parameters, this contour will be the pupil.

The eyebrow recognition follows a similar process, and was custom written by our group. Very few existing eye trackers that we could find dealt at all with eyebrows, and we think

eyebrow raise/lower recognition could provide useful control motions. The eyebrow recognition algorithm uses thresholding and contour finding, and then looks for the contour highest up in the image that is sufficiently large and oval-shaped.For this reason, it is important that the eye region be cropped short enough so that it doesn't contain other long/oval-shaped contours higher up (such as hair or the underside of the hat).

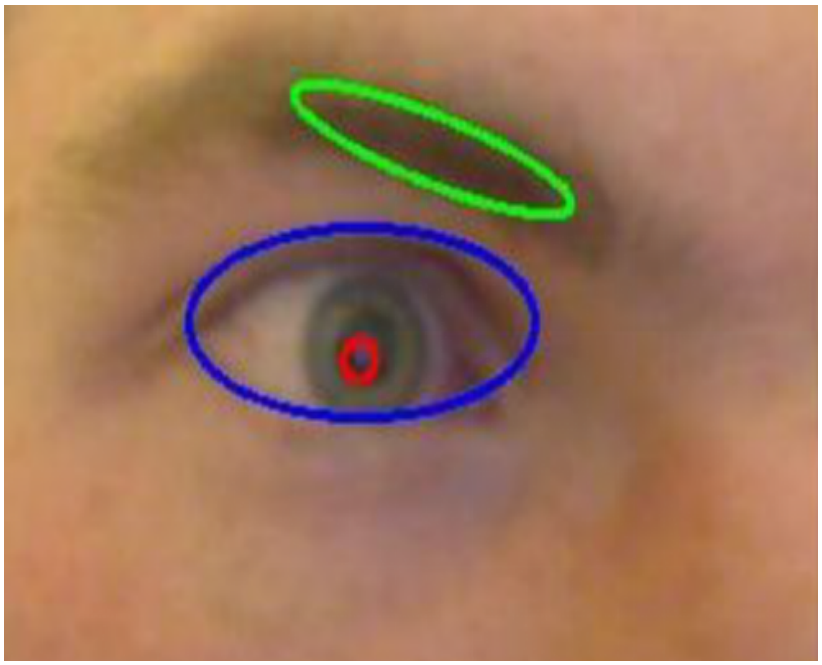

Figure: Tracking software showing it's best estimates of the pupil, eyebrow and eye locations

1.4.2. Gesture Recognition

Gesture Recognition's design was really a result of the limitations in Tracking. If tracking could be assumed to be perfect, Recognition would naturally be very trivial - similar to any system that tries to utilize real time, world information. Recognitions stated purpose is to ignore blinking, recognize winking, recognize eyebrow raising, recognize both eyes being closed deliberately, and recognizing deliberate eye movements left, right, up and down.

To achieve these operations, Recognition's goals therefore included: using temporal changes in the gaze to ignore outliers, filtering the pupil location over time to minimize

error, and using assumptions about the image and the effectiveness of tracking to improve recognition.

Initially we will discuss these goals then to establishing the parameters for recognition. Temporal Outliers: The camera operates at thirty hertz - quickly enough so that most actions occur over a series of frames. For example, natural blinking occurs over the course of 2 or 3 frames. We use that same time frame for a person to input a directional input (they must look upward for 3 frames).
Time Filtering: The pupil location and its radius are filtered over time with a Moving Average Filter. Tuning this over time revealed that storing three frames was sufficiently accurate. This is especially helpful for the radius, which can vary by two or three factors in bad tracking areas.This does not aid the tracking of eye position precisely. Rather, this is geared for when the user is trying to send a directional input: the added filtering improves the rate at which the software recognizes directional input, and helps minimize outliers skewing directional input.The recognition software was also set to be able to be recalibrated at any point, as well as receive extra calibration data at any point and integrate it into the running constants.

Assumptions Made: Since we are looking at both eyes simultaneously, we have an excess of data which can be whittled down to ensure the minimization of false positives and false negatives.One of our primary assumptions is that if a pupil is not recognized, the eye is considered to be blinking.Based on our own observations it was clear that the tracking software was better able to correctly recognize the pupil when the eye was looking inwards towards the center of the face (towards the other eye as opposed the nearest ear). Therefore, if either eye is seen looking inwards, that is taken as the correct gaze and the other is ignored. Furthermore, the tracking software tended to lose the pupil very easily if the eye was looking downwards, so the recognition software allows for gaps (by considering a blink during a series of downward eye actions to increase the count).With those features implemented, recognizing specific actions became relatively simple. The main recognizer loop was a function which updated after every image frame: the tracking software would call the Recognizer Class object's update loop, passing it the collected data for each eye.

The implementation used a finite state machine approach to go through every fame, adjusting the counts above to implement the temporal filtering and adding the necessary variables and adjusting the moving average filter.
The assumptions were implemented as part of the state machine. Variables that stored counts were used as temporal filters - actions had to continue for a prespecified number of frames before it would activate.

Once an action is determined, a keyboard event is sent, corresponding to the expectations used in testing. The eyebrow moving up would be sent as a 'w', right would be a 'd', eye winks were 'q' or 'e', a blink would b 'r' and an eyebrow raise would be 'f'.

1.5. PROPOSED MODULES

1.5.1 Gesture Recognition

Gesture Recognition's design was really a result of the limitations in Tracking. If tracking could be assumed to be perfect, Recognition would naturally be very trivial - similar to any system that tries to utilize real time, world information. Recognitions stated purpose is to ignore blinking, recognize winking, recognize eyebrow raising, recognize both eyes being closed deliberately, and recognizing deliberate eye movements left, right, up and down.
To achieve these operations, Recognition's goals therefore included: using temporal changes in the gaze to ignore outliers, filtering the pupil location over time to minimize error, and using assumptions about the image and the effectiveness of tracking to improve recognition.

1.5.2 Feature-Based Techniques

Feature-based techniques are based on the identification and utilization of a set of unique features of the human eyes. These techniques identify such local features of the eye and the face which have reduced sensitivity to variations in viewing angles and illumination. The commonly used features for eye localization are corneal reflections, limbus, and dark and bright pupil images. Typically, these techniques first identify and detect the local features; then, they apply a filter to highlight desired features while suppressing the others

or utilize a prior eye shape model to construct a local contour; and, finally, they apply the classification algorithms to produce the output. Generally, the feature-based techniques are reported to provide good results in indoors applications; however, their outdoor performance is comparatively limited. These techniques are further subcategorized as follows.

1.5.3 Detection of Iris and Pupil

The pupil and iris being darker than their surroundings are commonly considered reliable features for eye detection. We can use a skin-color model and introduce an algorithm to locate the pupils by searching for two dark areas that fulfill specific anthropometric requirements. Their technique,however, cannot perform well in dierent light conditions due to limitation of the skin-color model.

Generally, use of IR light instead of visible light seems more appropriate for dark region detection. The techniques based on iris and pupil detection require the images taken from close to the eyes or high-resolution images.The majority of the feature-based techniques cannot be used to model closed eyes. In an eort to overcome this limitation, a method can be proposed to track the eyes and to retrieve the eye parameters with the help of a dual-state (i.e., open or closed) eye model. The eyelids and eyes' inner corners are detected through the algorithm proposed in This technique, however, requires a manual initialization of the eye model and high contrast images.

The modeling of gaze direction is based either on the visual axis or on the optical axis. The visual axis, which forms the line of sight (LoS) and is considered the actual direction of gaze, is the line connecting the center of the cornea and the fovea. The optical axis, or the line of gaze (LoG), is the line passing through the centers of pupil, cornea, and the eyeball. The center of cornea is known as the nodal point of the eye.

The visual and optical axes intersect at the nodal point of the eye with a certain angular o set. The position of head in 3D space can be directly estimated by knowing the 3D location of the corneal or eyeball center. In this way, there remains no need for separate head

location models. Thus,the knowledge of these points is the keystone for majority of the head pose invariant models.The objective of gaze tracking process is to identify and track the observer's point of regard (PoR) or gaze direction.

For this purpose, the important features of eye movements such as fixation, saccades, and smooth pursuit are utilized. Fixation represents the state when the observer's gaze rests for a minimum time (typically more than 80–100 ms) on a specific area within 2–5 of central vision.Saccades are quick movements of eyes that take place when visual attention transfers between two fixed areas, with the aim of an bringing area of interest within the narrow visual field. When a driver visually follows a traveling object, this state is represented by smooth pursuit . The data associated with the fixations and saccades provides valuable information that is used for the identification and classification of vision, neurological, and sleep conditions. In the field of medical psychology, data of the fixations is utilized to analyze a person's attentiveness and level of concentration.

Saccadic eye movements are widely studied in a variety of applications such as human vision research and drowsiness detection for vehicle drivers. Moreover, saccade is also used as a helpful index for determination of mental workload. Studies show that the saccade distance decreases when the task's complexity increases .

1.6. HARDWARE AND SOFTWARE REQUIREMENTS

1.6.1. Hardware Components

Different hardware and network resources may be available or desired for a given application. The common architectural components are:

- Operating System: Windows XP/7/8/8.1/10
- RAM: 4 GB
- Hard Disk: 256 GB SSD
- Processor: Intel Dual Core or higher processor

Persistent server / desktop – low quantity, high cost, high processing power and memory. These systems will typically have server grade x64 processors and potentially GPU processors.

Embedded device – low-cost, high quantity devices with limited processing power and memory that can either host FR libraries on-edge or operate as a “thin-client” that passes imagery to a server or cloud system for processing. These systems typically have mobile grade ARM processors and potentially Neural Processing Units (NPU’s).
Scalable cloud – arrays of server resources abstracted through a cloud resource management system
Network – communication channels between devices. Networks will have varying amounts of bandwidth depending on their properties

1.6.2. SOFTWARE REQUIREMENTS

1. VISUAL STUDIO

Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs including websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works as both a source-level debugger and as a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new tool sets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

2. SQL LITE 3

SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. SQLite is a compact library. With all features enabled, the library size can be less than 750KiB, depending on the target platform and compiler optimization settings. (64-bit code is larger. And some compiler optimizations such as aggressive function inlining and loop unrolling can cause the object code to be much larger.) There is a tradeoff between memory usage and speed. SQLite generally runs faster the more memory you give it We the developers hope that you find SQLite useful and we entreat you to use it well: to make good and beautiful products that are fast, reliable, and simple to use. Seek forgiveness for yourself as you forgive others. And just as you have received SQLite for free, so also freely give, paying the debt forward.

3. INNOSETUP

InnoSetup is an open source compiler to create installers on windows. It is free and provides a rich feature set. You can create professional looking installers for the end user. In this article we will explain you the basic concepts and steps involved in creating a simple setup of your Project. InnoSetup script file is a simple text file which is similar to .INI files with the extension .ISS. In this script file the contents are arranged in sections. These scripts are easy to understand and uses a simple syntax. The section starts with the Section Name which is enclosed in square brackets.

4. PYINSTALLER

PyInstaller reads a Python script written by you. It analyzes your code to discover every other module and library your script needs in order to execute. Helps in our project , Then it collects copies of all those files – including the active Python interpreter! – and puts them with your script in a single folder, or optionally in a single executable file.

5. DB Browser for SQLLITE

It is a tool that lets us view the data that is stored in an SQLite Database. Depending on the format and type of data in the database it may or may not be readable by a human. This is generally used for debugging or other development tasks where the developer needs to read the data that has been stored but does not have a built-in system to access it through the program.

6. PYTHON Version 3

Python has numerous uses. Most commonly it is applied to web application development, particularly when you want the process to move quickly. Frameworks like speed development along. Python is also a player in the data science world. Professionals in the field take advantage of the language's ability to facilitate complex calculations and mine, processes, and build visualizations, thanks to features like its simplicity and straightforward syntax. Data scientists and analysts often leverage Python in their work because of these qualities. As machine learning, a subfield of artificial intelligence, gains traction, Python is on the rise among developers in this field, too. In fact, the vast majority of **machine learning** developers and specialists use the language. Which was the base for our project. We uses different libraries like numpy, pandas, imutils, pyautogui, pillow, Tkinter, Os, Sys, Cv2, sqlite3, pymysql, pyrebase, dlib, wincertstore, six, piglet, certifi .

CHAPTER 2
SYSTEM ANALYSIS AND SPECIFICATION

2.1. Functional Model

Functional Modelling provides the outline that what the system is supposed to do. It does not describe what is the need of evaluation of data, when they are evaluated and how they are evaluated apart from all it only represents origin of data values. It describes the function of internal processes with the help of DFD (Data Flow Diagram).

Data Flow Diagrams: Function modelling is represented with the help of DFDs. DFD is the graphically representation of data. It shows the input, output and processing of the system. When we are trying to create our own business, website, system, project then there is need to find out how information passes from one process to another so all are done by DFD.

2.1.1 Data Flow Diagram

In Software engineering DFD (data flow diagram) can be drawn to represent the system of different levels of abstraction. Higher-level DFDs are partitioned into low levels-hacking more information and functional elements. Levels in DFD are numbered 0, 1, 2 or beyond. Here, we will see mainly 3 levels in the data flow diagram, which are: 0-level DFD, 1-level DFD, and 2-level DFD.

0-level DFD:

It is also known as a context diagram. It's designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.

In this 0-level DFD we have used 2 external entities named user and system which are the basic modules of our project. And the data process here used is our system controller

software module it takes the input from the User Entity in the form of gestures and the system controller will recognize the gesture and gives the input to the system, then system will detect the acquired signals to the controller.

Then the output is the given task that is conveyed by the user to our system software controller is performed efficiently.

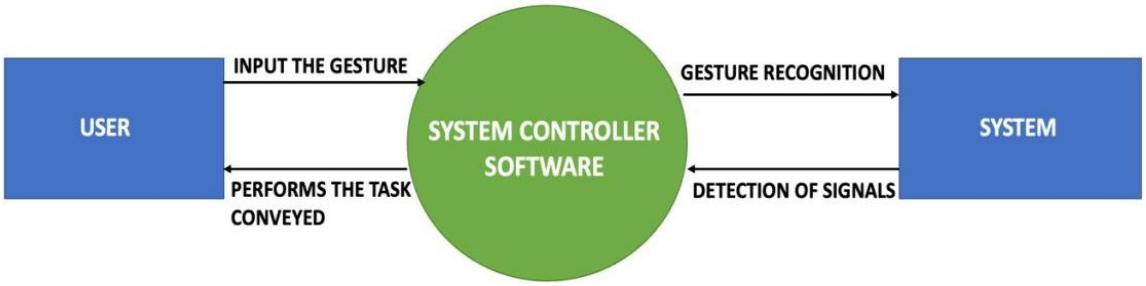

FIGURE 2.1 : 0- LEVEL DFD

1-level DFD:

In 1-level DFD, the context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.

In 1-level DFD, depicted above we have used a single external entity which is the designated user and 4 external processes named the System Controller Software, Gesture Recognition Processes, Eye Detection Module, Report to The System Module. The activities used here are gaze co-ordinates, gaze estimation, estimated head pose. And the data process here used is our system controller software module it takes the input from the User Entity in the form of gestures and the system controller will recognize the gesture and gives the input to the system, then system will detect the acquired signals to the controller. The eye detection module takes and give input to the controller software as it works on the detecting of eye gestures and hence produces 2 activities which constitutes estimated head pose and gaze estimation. Then in gesture recognition module, the gesture recognition processes work simultaneously with the eye detection module and gives the desired input and output to the controller software which makes it work efficiently.

Finally, the gathered output is then reported to the system and hence the result is conveyed to the user and the given task is performed.

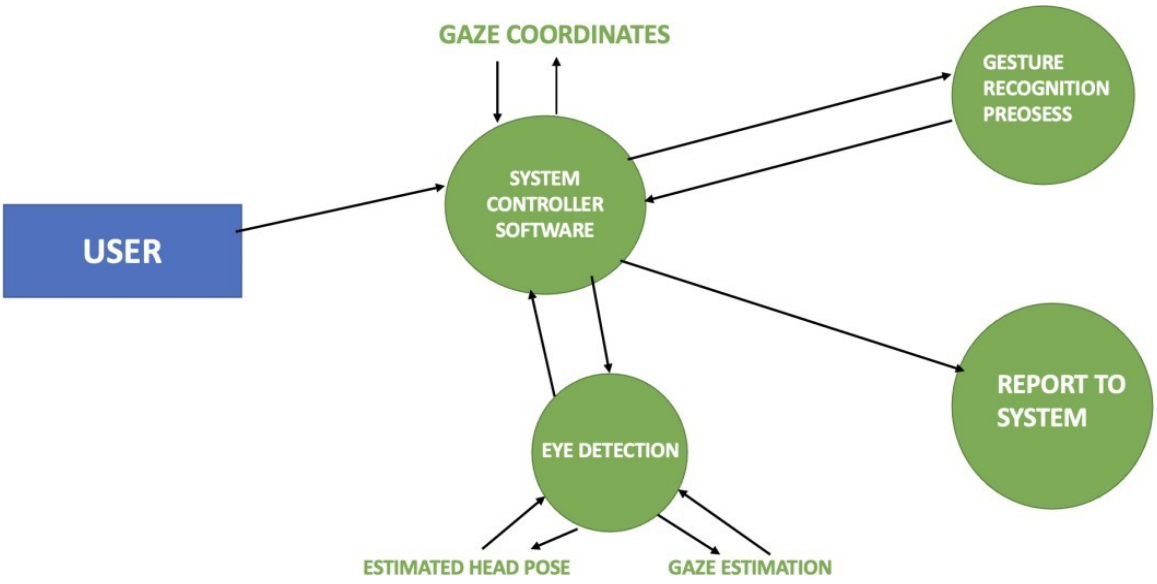

Figure 2.2: 1- Level DFD

2-level DFD:

2-level DFD goes one step deeper into parts of 1-level DFD. It can be used to plan or record the specific/necessary detail about the system's functioning.

In this 2-level DFD, we have used 3 external entities named user, report, output and the 9 external processes here used are Query Process, System Controller Software, Enquiry, Eye Detection, Eye Tracking, Searching/ Calibration, Gesture Recognition Processes, Perform Action and Report to The System and the designated activities used here are gaze estimation, estimated head pose, gaze co-ordinates, eye position and gaze co-ordinates. And the data process here used is our system controller software module it takes the input from the User Entity in the form of gestures and the system controller will recognize the gesture and gives the input to the system, then system will detect the acquired signals to the controller. The eye detection module takes and give input to the controller software as

it works on the detecting of eye gestures and hence produces 2 activities which constitutes estimated head pose and gaze estimation. Then in gesture recognition module, the gesture recognition processes includes the searching/calibration external process which further includes recognition of the users eye according to the screen co-ordinates which will be calibrated with the eyes of the user which will help the software to calculate the correct co-ordinates on the screen on which the user is gazing .Also, the eye detection module along with the eye tracking module works simultaneously with the system controller software module and gives the desired input and output to the system which makes it work efficiently. Finally the gathered output is then reported to the system and hence the result is conveyed to the user and the given task is performed. If any query arises then the query process module will be activated and will work efficiently to resolve the query and then the result will be constituted to the report module.

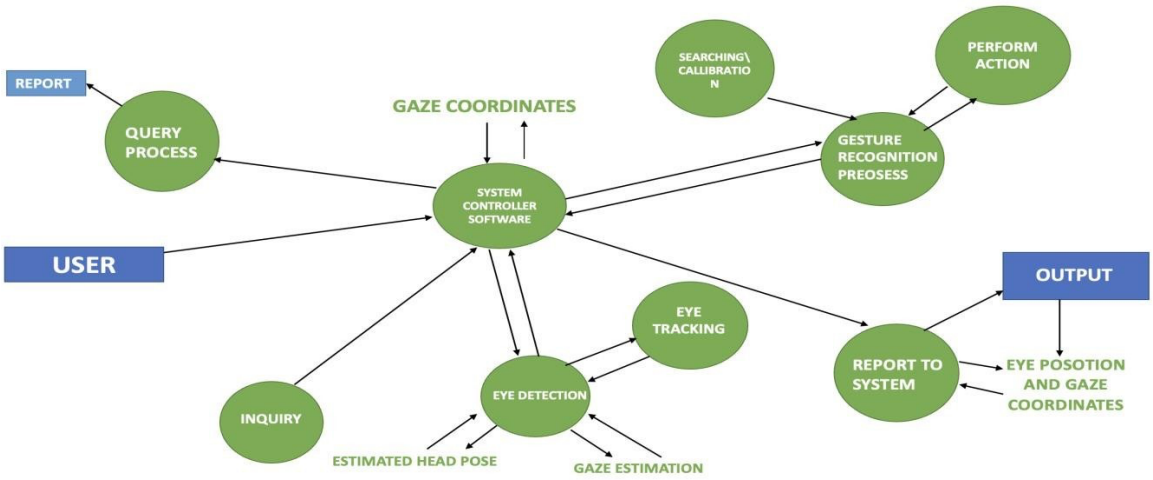

Figure 2.3: 2- Level DFD

2.1 DATA MODEL

Data models are visual representations of an enterprise's data elements and the connections between them. By helping to define and structure data in the context of relevant business processes, models support the development of effective information systems. Data models play a key role in bringing together all segments of an enterprise – IT, business analysts,

management and others – to cooperatively design information systems (and the databases they rely on). These systems require properly defined and formatted data, and models shine a clear light on what data is required and how it must be structured to support the desired business processes. By explicitly determining the structure of your data, these models support a variety of use cases, including database modeling, information system design, and process development in support of a consistent, clean exchange of data. It's also important to understand the three different types of data models. Each will serve a different purpose as you work through the data modeling process.

2.2.1. ER DIAGRAM

Entity-Relationship Data Model: An ER model is the logical representation of data as objects and relationships among them. These objects are known as entities, and relationship is an association among these entities. This model was designed by Peter Chen and published in 1976 papers. It was widely used in database designing. A set of attributes describe the entities. For example, student_name, student_id describes the 'student' entity. A set of the same type of entities is known as an 'Entity set', and the set of the same type of relationships is known as 'relationship set'.

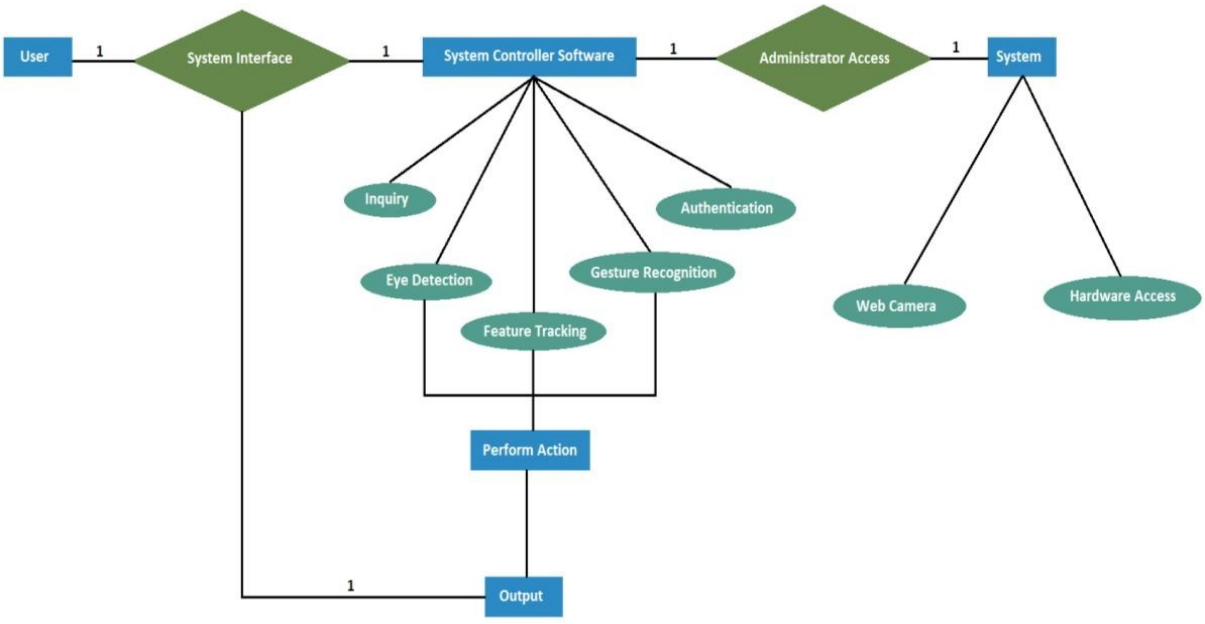

Figure 2.2.1.1: ER Diagram

In this ER Diagram the User firstly access the system interface which will give the access to the software after running the software it will ask for the administrator permission from the user which will grant it the access to the hardware as well as the system software. In which in Hardware Permission it will take the admin permission foe Camera and Screen capturing. Then after successful Access to both hardware it will run its different modules as Inquiry, Eye Detection, Face Tracking, Gesture Recognition, Authentication, After the processing of Data received by the different modules it will act accordingly and give output to the system Interface.

2.3. A Process-Flow Model

Workflow-

In general workflow refers to the series of sequential tasks those are performed to achieve certain goal. Each workflow step is defined by three parameters i.e input, transformation, and output. In workflow process a series of actions are performed to achieve a business outcome.

Software-Process-Workflows-

Software process is the set of related activities those are performed to get a software product as an outcome and there the software process workflows leads the software developments in a linear way by performing series of sequential tasks.

2.3.1 Activity Diagram

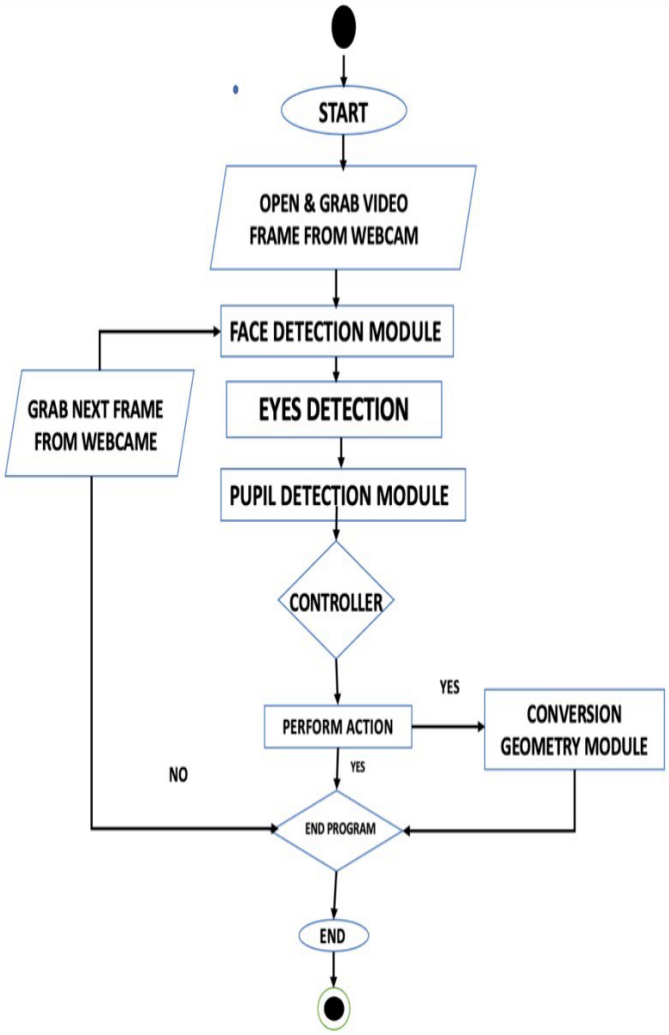

Figure 2.3.1.1: Activity Diagram

We use Activity Diagrams to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on condition of flow and the sequence in which it happens.

In this activity Diagram after starting the executable file it will capture the real time data from the system hardware using Screen and Web Cam, from Which the data flows to the

Face Detection Module, Eye Detection Module, Pupil Detection Module after Processing the data the controller performs the desirable action using Geometric Conversion Module then end the Current Operation and Again take the data at real time and works till the Controller is Running.

2.4. Behavioral Model

Behavioural Model is specially designed to make us understand behaviour and factors that influence behaviour of a System. Behaviour of a system is explained and represented with the help of a diagram. This diagram is known as State Transition Diagram. It is a collection of states and events. It usually describes overall states that a system can have and events which are responsible for a change in state of a system. So, on some occurrence of a particular event, an action is taken and what action needs to be taken is represented by State Transition Diagram. Behavioral modeling is an approach used by companies to better understand and predict consumer actions. Behavioral modeling uses available consumer and business spending data to estimate future behavior in specific circumstances.

2.4.1. Sequence Diagram

A sequence diagram is the most commonly used interaction diagram. Interaction diagram – An interaction diagram is used to show the interactive behavior of a system. Since visualizing the interactions in a system can be a cumbersome task, we use different types of interaction diagrams to capture various features and aspects of interaction in a system.

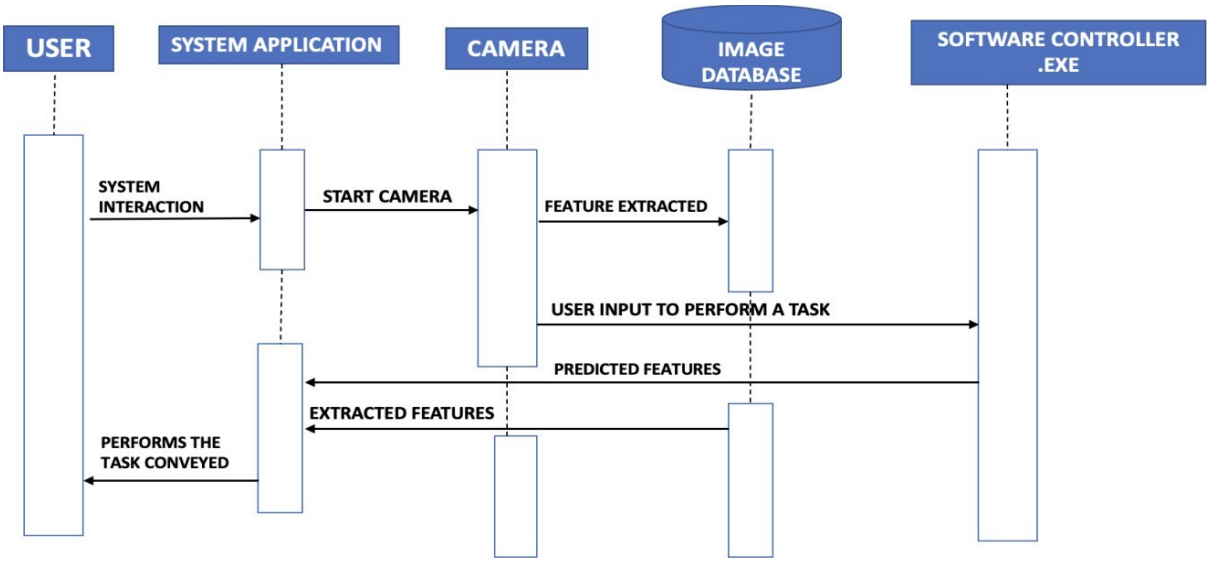

Figure 2.4.1.1: Sequence Diagram

In this Sequence Diagram we represented the flow of the interaction of the different processes / modules.

In this the User access the system controller which then starts the camera which further captures the Eye/ facial gesture which will be then compared with the data in the image database after which the system controller.exe will calculate the data and predict the action and then it will perform the action and gives the desired output to the user.

2.5. System Design

2.5.1. Technical feasibility

1. The eye gesture control system directly interacts with the vision of the human eyes and then controls the system.
2. Eye gesture, a real-time gesture assurance programming which controls a computer mouse cursor by using the user's eye gestures.
3. By looking at the control of a system that is display on a screen, a person can run a computer software.

2.5.2. Operational feasibility

1. The system will be implemented into the PC devices which may be based on different operating systems. Also, built in web cameras and a powerful processor are needed for the system.
2. Since the system will work on real time, performance is one of the most important topics of the system. Performance of the system will be high enough that user can use the system without noticeable delays or performance problems.
3. Software will work on real time. Response of the system after the user makes an action will be fast enough that does not cause a problem.

2.5.3. Economic feasibility

1. Implementation of the stereovision technology by using web camera provides an inexpensive solution with respect to the current technologies which are already available on the personal computer device market.
2. Any error occurred while the system is in use will be tolerated and system recovers itself and continues properly.

CHAPTER 3
MODULE IMPLEMENTATION AND SYSTEM INTEGRATION

3.1. Module Implementation

3.1.1User

This is the First module from which the user will be able to ace

ss the all-different types of features and the modules in the software application. Here There will be a Walk through for the Setting Up the Software and for a hassle- free setup and user experience.

3.1.2. Sign Up

When user interacts with the module it comes up with a sign in page. Here he/she must sign up first to register their details and all the information which is required.

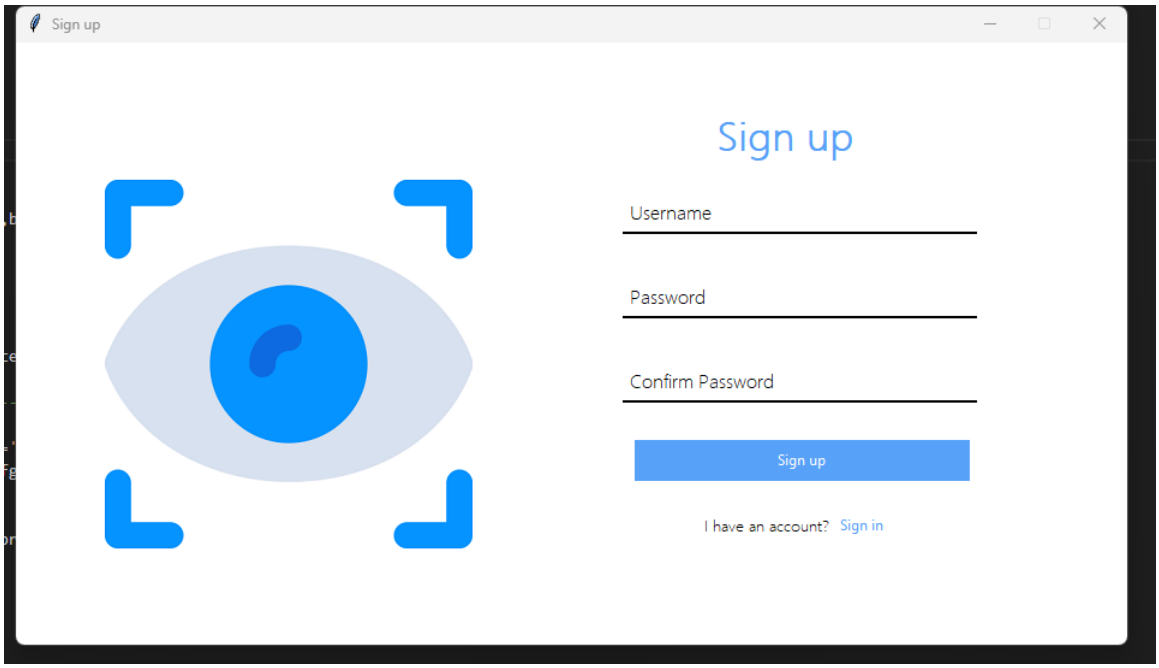

Figure: 3.1.2.a Sign UP Page

3.1.3. Sign In

After signing up the user can now able to login or sign in using his/her user name and password which was registered by the user at the time of signing up.

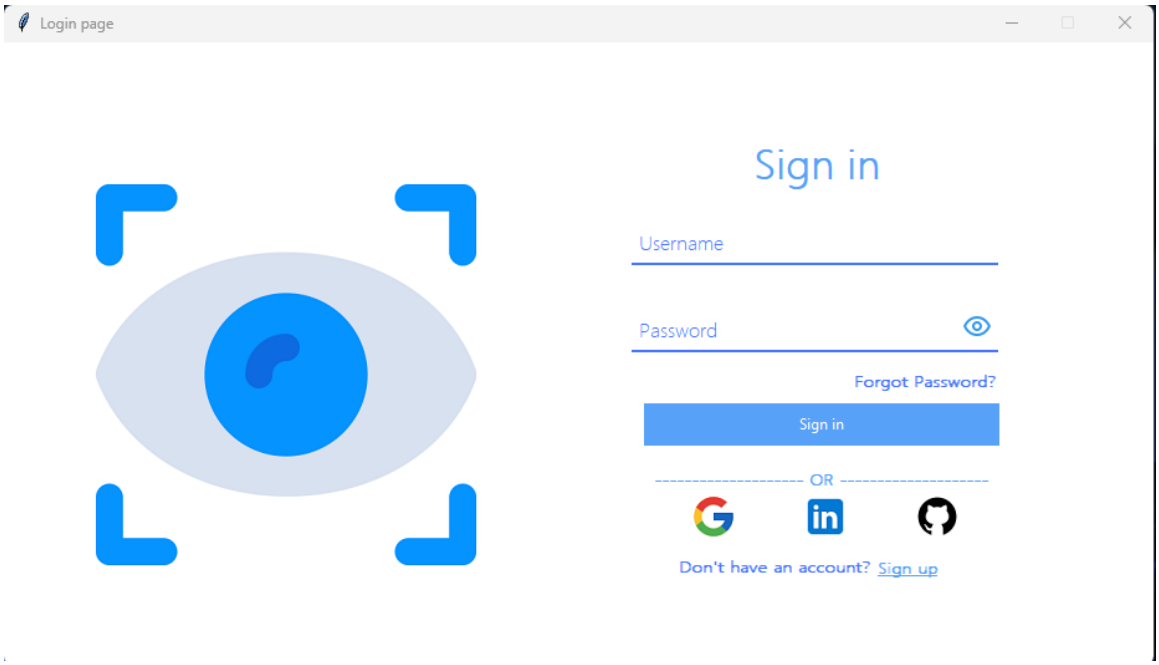

Figure: 3.1.3.a Sign In Page

3.2. System Controller

1. The eye gesture control system directly interacts with the vision of the human eyes and then controls the system. Eye gesture, a real-time gesture assurance programming which controls a computer mouse cursor by using the user's eye gestures. Only requirement to operate the imouse system is, individuals having at least one eye with good vision and ability to control the computer. Its users can be grown-ups and youngsters with cerebral paralysis, spinal rope wounds, mental wounds, ALS, different sclerosis, Brainstem strokes, and so on.

2. Eye gesture control system can be utilized as a part of homes, workplaces, schools, healing centers, and long-haul mind offices. By looking at the control of a system that is display on a screen, a person can run a computer software, operate a

computer mouse, and access the internet and also email. Provide a cheap eye-tracking system, to control the cursor of a computer with eyes. Allow physically disabled people to use computers

3. To control a computer and communicate with other systems. To provide a real-time accurate eyes gesture control system. To provide a hand free mouse control system. To provide a complete generic eye-gesture mouse control system. To provide a complete wire free mouse control system. Easy to control cursor movement of a mouse.

4. Eye gesture control system can be utilized as a part of homes, workplaces, schools, healing centers, and long-haul mind offices. By looking at the control of a system that is display on a screen, a person can run a computer software, operate a computer mouse, and access the internet and also email. Provide a cheap eye-tracking system, to control the cursor of a computer with eyes. Allow physically disabled people to use computers. To control a computer and communicate with other systems. To provide a real-time accurate eyes gesture control system. To provide a hand free mouse control system. To provide a complete generic eye-gesture mouse control system.

5. To provide a complete wire free mouse control system. Easy to control cursor movement of a mouse. Going with the literature, we are focusing on customer satisfaction who are physically disabled. In 2018, an eye tracking algorithm based on Hough transform was developed. This system detects the face and eyes of a person. It uses a webcam to detect user's face and eyes. The issue in this system is of real-time tracking and time speed issue. The system is quite slow and it needs a high-quality computer system to work properly which is costly.

3.3. Eye Detection

Detection of Iris and Pupil

The pupil and iris being darker than their surroundings are commonly considered reliable features for eye detection. We can use a skin-color model and introduce an algorithm to locate the pupils by searching for two dark areas that fulfill specific anthropometric requirements. Their technique, however, cannot perform well in direct light conditions due to limitation of the skin-color model. Generally, use of IR light instead of visible light seems more appropriate for dark region detection. The techniques based on iris and pupil detection require the images taken from close to the eyes or high-resolution images. The majority of the feature-based techniques cannot be used to model closed eyes. In an sort to overcome this limitation, a method can be proposed to track the eyes and to retrieve the eye parameters with the help of a dual-state (i.e., open or closed) eye model. The eyelids and eyes' inner corners are detected through the algorithm proposed in This technique, however, requires a manual initialization of the eye model and high contrast images

Gaze Tracking

The typical eye structure used in gaze tracking applications is demonstrated in Figure. The modeling of gaze direction is based either on the visual axis or on the optical axis. The visual axis, which forms the line of sight (LoS) and is considered the actual direction of gaze, is the line connecting the center of the cornea and the fovea. The optical axis, or the line of gaze (LoG), is the line passing through the centers of pupil, cornea, and the eyeball. The center of cornea is known as the nodal point of the eye. The visual and optical axes intersect at the nodal point of the eye with a certain angular o set. The position of head in 3D space can be directly estimated by knowing the 3D location of the corneal or eyeball center. In this way, there remains no need for separate head location models. Thus, the knowledge of these points is the keystone for majority of the head pose invariant models.

The objective of gaze tracking process is to identify and track the observer's point of regard (PoR) or gaze direction. For this purpose, the important features of eye movements such as fixation, saccades, and smooth pursuit are utilized. Fixation represents the state when the observer's gaze rests for a minimum time (typically more than 80–100 ms) on a specific

area within 2–5 of central vision. Saccades are quick movements of eyes that take place when visual attention transfers between two fixed areas, with the aim of a bringing area of interest within the narrow visual field. When a driver visually follows a traveling object, this state is represented by smooth pursuit. The data associated with the fixations and saccades provides valuable information that is used for the identification and classification of vision, neurological, and sleep conditions. In the field of medical psychology, data of the fixations is utilized to analyze a person's attentiveness and level of concentration. Saccadic eye movements are widely studied in a variety of applications such as human vision research and drowsiness detection for vehicle drivers. Moreover, saccade is also used as a helpful index for determination of mental workload. Studies show that the saccade distance decreases when the task's complexity increases.

3.4. Gesture Recognition

Gesture recognition provides real-time data to a computer to make it fulfill the user's commands. Motion sensors in a device can track and interpret gestures, using them as the primary source of data input. A majority of gesture recognition solutions feature a combination of 3D depth-sensing cameras and infrared cameras together with machine learning systems. Machine learning algorithms are trained based on labeled depth images of hands, allowing them to recognize hand and finger positions. We wish to present a novel method of human-computer interaction in this technology that is based on eye and facial gestures. It is software that uses an image sensing device to communicate with gestures. In order to synthesize natural gestures, acts gesture and symbol gesture were compared. Using Data- Driven Gesture Synthesis: Evaluating Gesture Units Hypoglossal facial anastomosis and jump interposition graft in the treatment of facial reanimation: Evolution The initial goal of the software is
,in the first time to detect user's face and their eyes in the second time. The eyes gesture recognition is based on fuzzy inference system. The fuzzy inference system is the foundation for the eyes gesture recognition.

Gesture recognition consists of three basic levels:

1.Detection:

With the help of a camera, a device detects hand or body movements, and a machine learning algorithm segments the image to find hand edges and positions.

2. Tracking:

A device monitors movements frame by frame to capture every movement and provide accurate input for data analysis.

3. Recognition:

The system tries to find patterns based on the gathered data. When the system finds a match and interprets a gesture, it performs the action associated with this gesture. Feature extraction and classification in the scheme below implements the recognition functionality. Keep in mind that hand tracking and gesture recognition are not the same things. Both technologies are supposed to use hands for human-machine interaction (HMI) without touching, switching, or employing controllers. Sometimes, systems for hand tracking and gesture recognition require the use of markers, gloves, or sensors, but the ideal system requires nothin but a human hand. Why may people want to use gestures instead of just touching or tapping a device?

A desire for contactless sensing and hygiene concerns are the top drivers of demand for touchless technology. Gesture recognition can also provide better ergonomics for consumer devices. Another market driver is the rise of biometric systems in many areas of people's lives, from cars to homes to shops. During the coronavirus pandemic, it's not surprising that people are reluctant to use touchscreens in public places. Moreover, for drivers, tapping a screen can be dangerous, as it distracts them from the road. In other cases, tapping small icons or accidentally clicking on the wrong field increases frustration and makes people look for a better customer experience. Real- time hand gesture recognition for computer interactions is just the next step in technological evolution, and it's ideally suited for today's consumer landscape.

Besides using gestures when you cannot conveniently touch equipment, hand tracking can be applied in augmented and virtual reality environments, sign language recognition, gaming, and other use cases. The high cost of touchless sensing products is one of the major challenges of this technology, along with the complexity of software development for HGR. To create a robust system that detects hand positions, a hand tracking solution requires the implementation of advanced machine learning and deep learning algorithms, among other things.

3.5. Perform Action

This modules will perform the final output which id given by the System Controller.exe and performs the output on the system Hardware and perform the desired task.

3.6. System Integration

System integration is the process of connecting different sub- system(components) or modules into a single larger system or program that function as one. With regards to software solution, system integration is typically defined as the process of linking together various modules. Systems integration consists of assuring that the pieces of a project come together at the right time and that it then functions as an integrated unit. However, to accomplish the integration process, all the various types of interfaces must be monitored and controlled, because integration, for the most part, is just another way of saying interface management. In addition, the number of interfaces can increase exponentially as the number of organizational unit's increase.

3.7.Landing Page

A landing page is a standalone page designed to communicate a well-defined message and persuade visitors to perform a specific desired action. That action might be joining an email list, signing up for a free product trial, downloading a whitepaper, buying a product, or clicking a call-to-action (CTA) button.

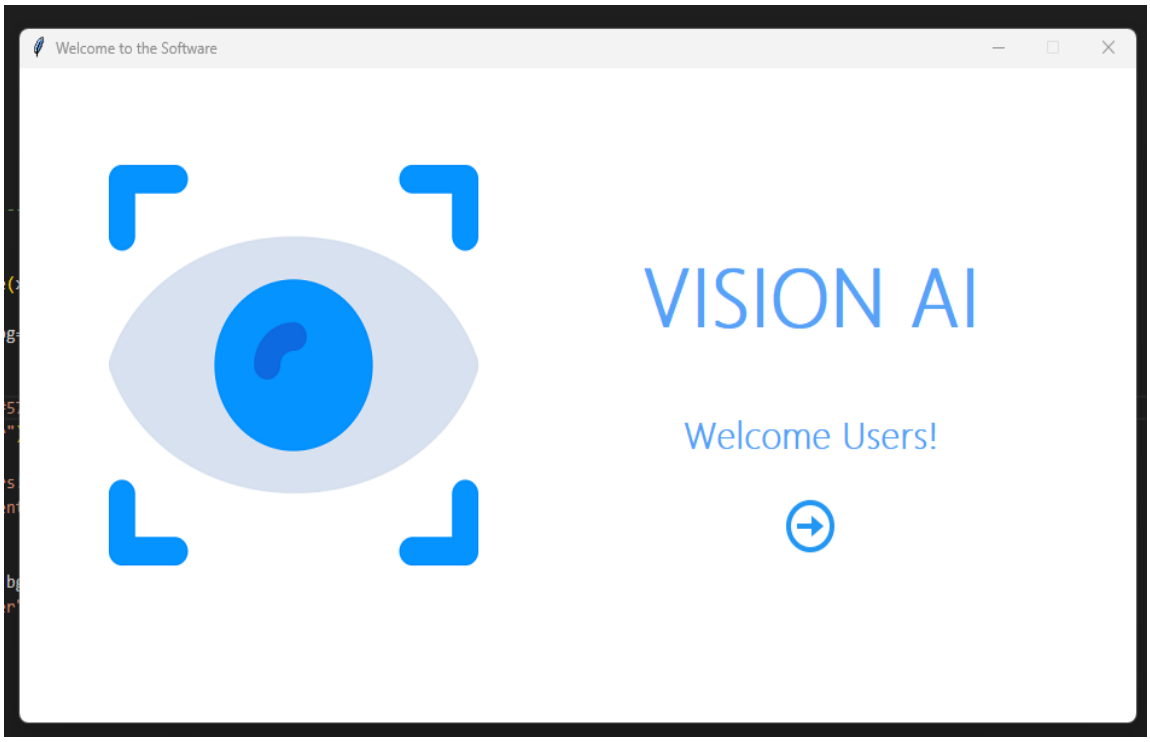

Figure: 3.7.a Vision AI Page

3.8. Camera Accessibility

We uses camers accessibility as we need to access the camera for capturing the eye and facial gestures of the user to operate the system as per his/her use.Firstly, go to setting to allow the permission for accessing the camera.

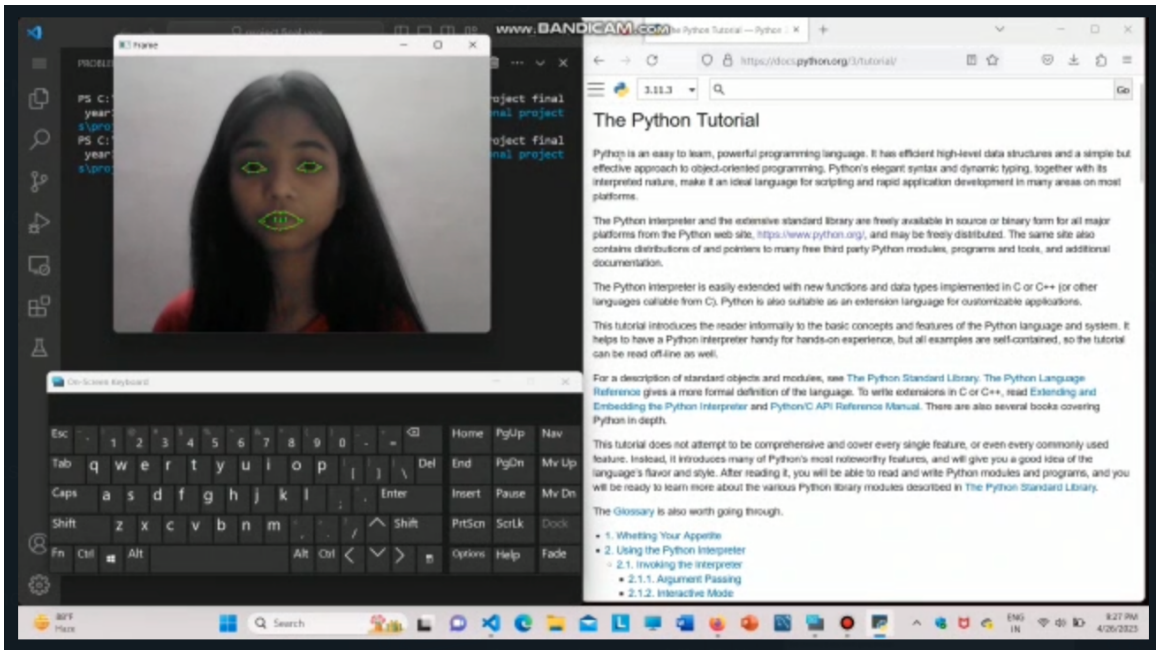

Figure: 3.8.a C Accessibility

3.8.1. Reading Mode

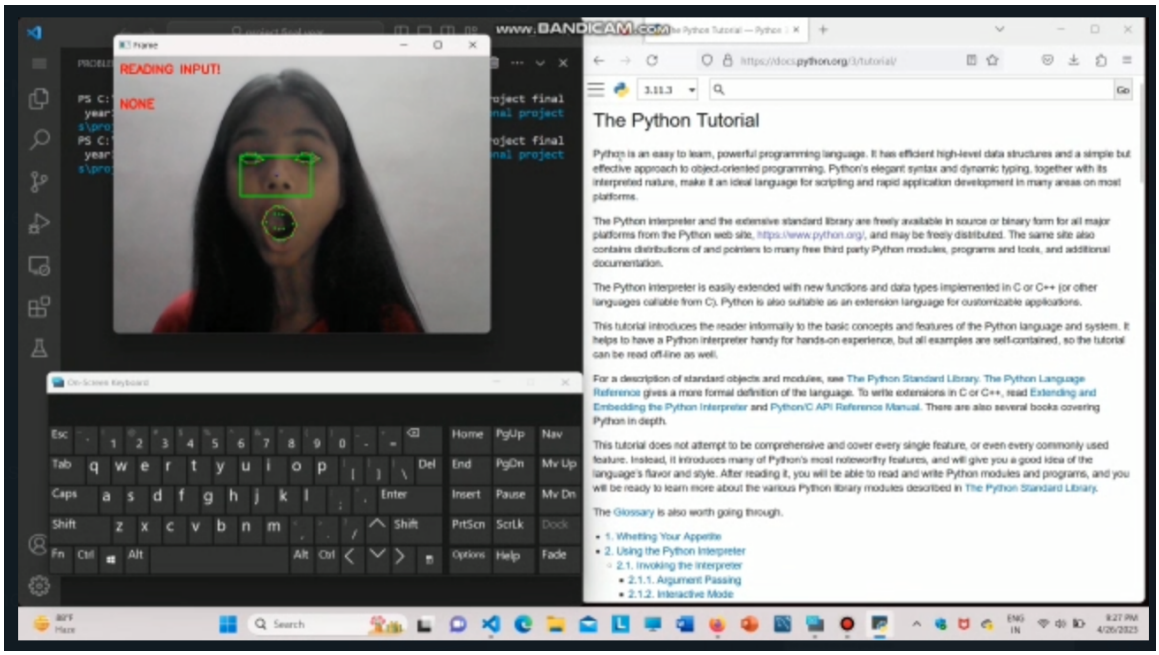

Figure: 3.8.1.a Reading Mode

3.8.2. Reading Mode to Left

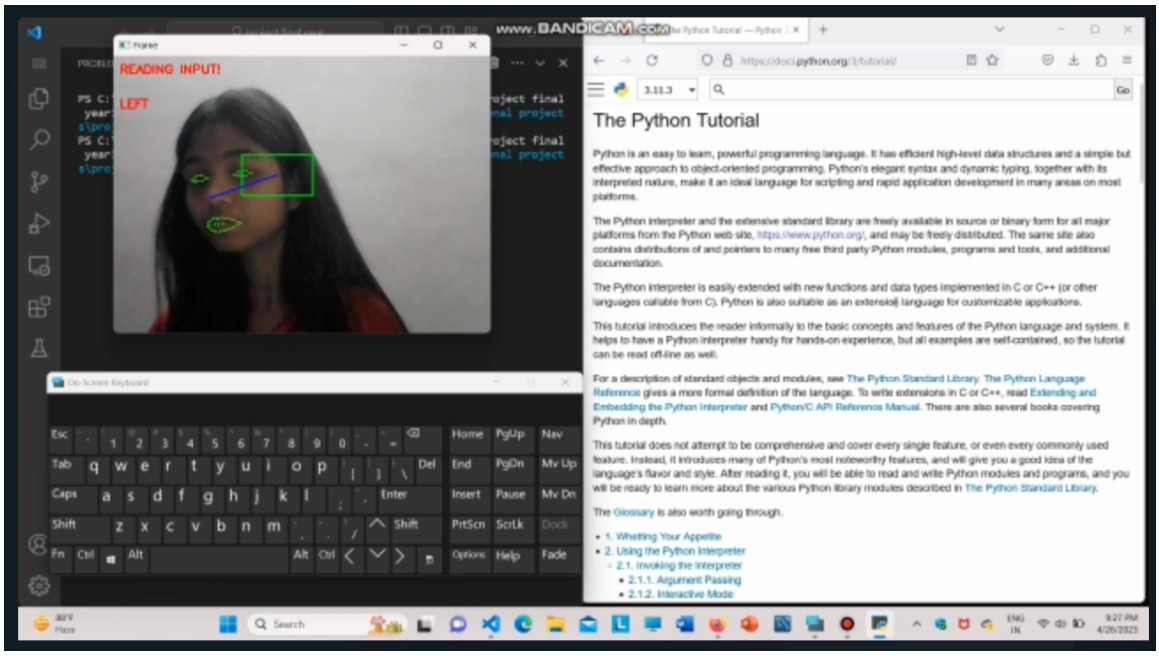

Figure: 3.8.2.a Left Read

3.8.3. Reading Mode to Right

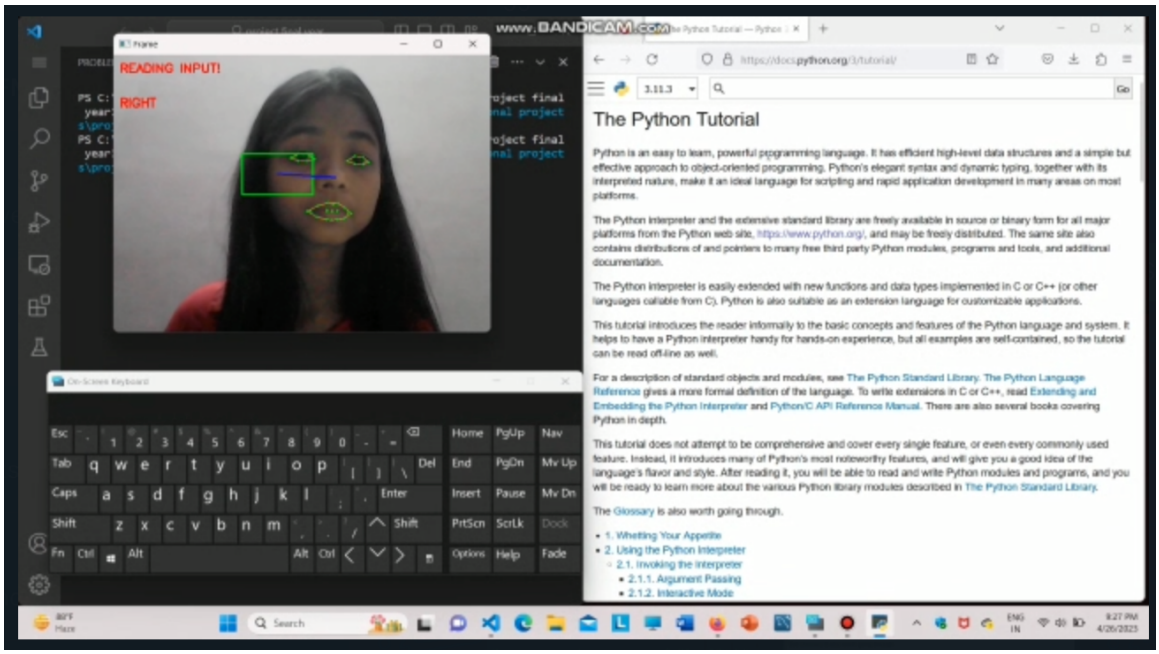

Figure: 3.8.3.a Right Read

3.8.4. Reading Mode to Down

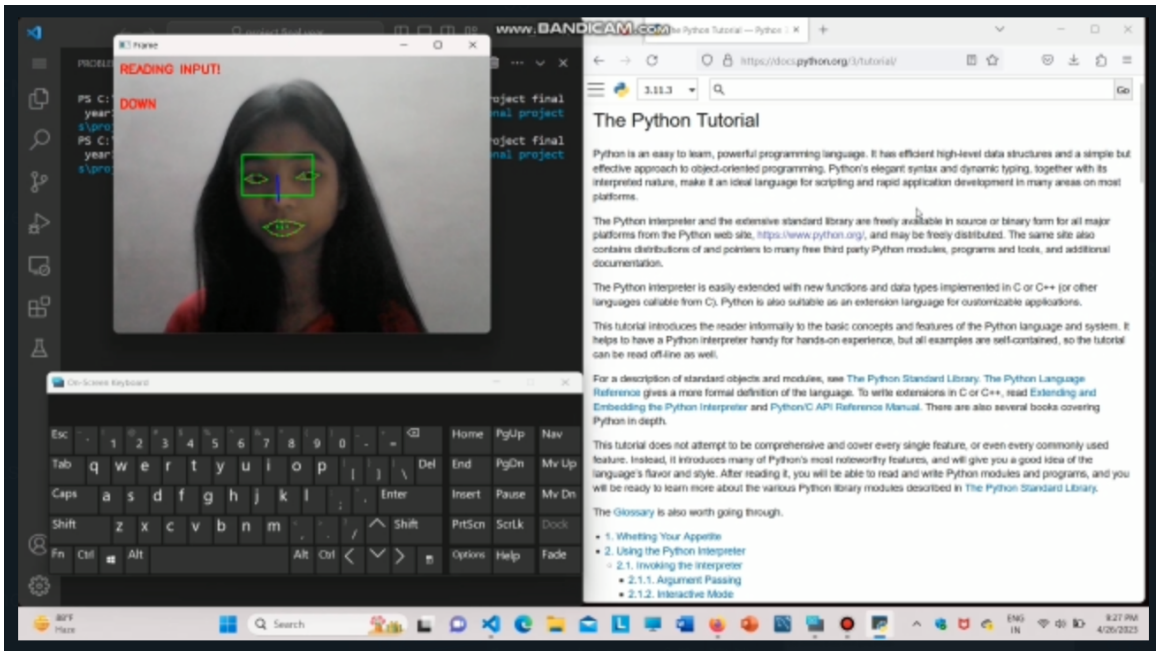

FIGURE: 3.8.4.a Down

3.8.5. Reading Mode to Up

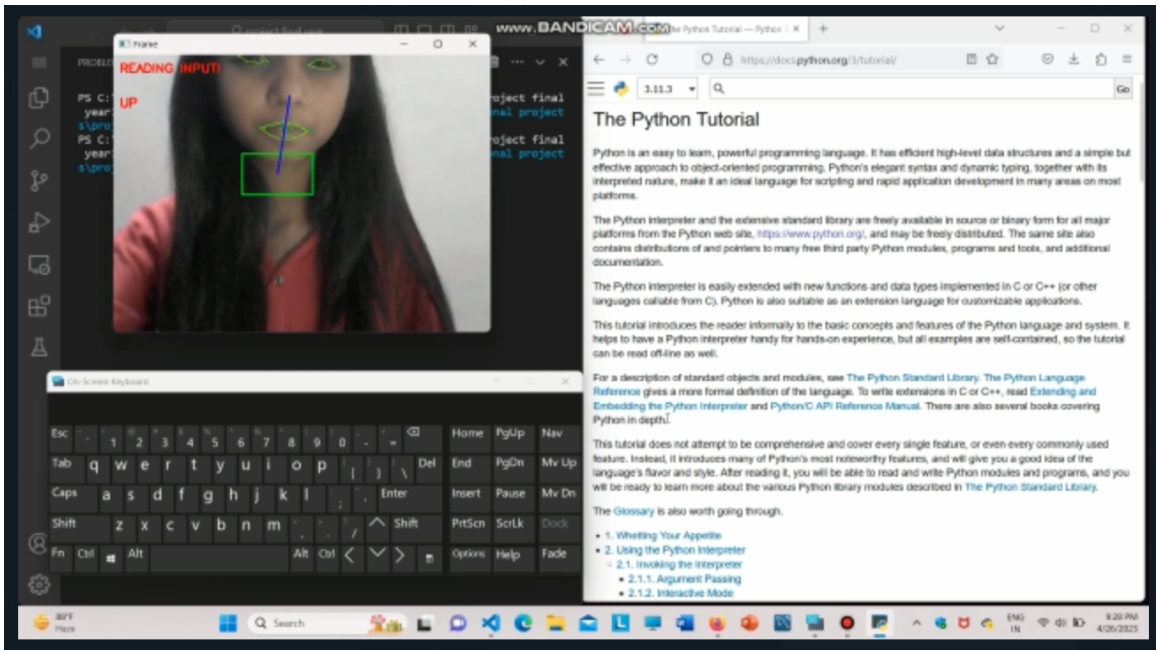

FIGURE: 3.8.5.a Up Read

3.9. Scrolling Mode

We uses camers accessibility as we need to access the camera for capturing the eye and facial gestures of the user to operate the system as per his/her use to scroll the mouse.

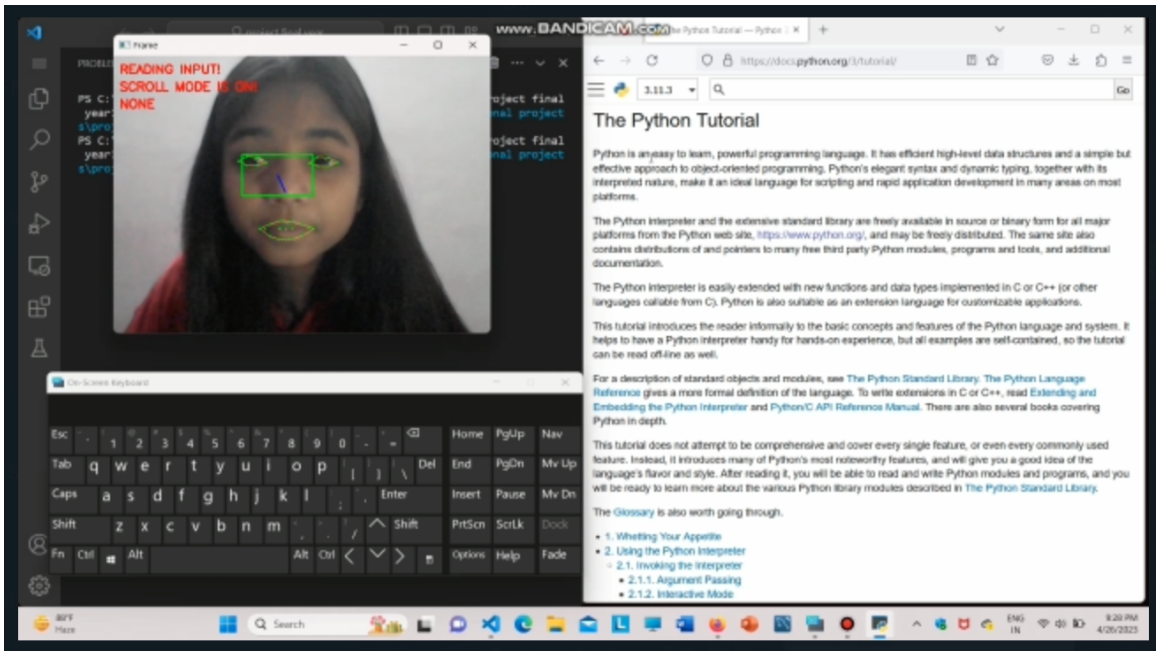

Figure: 3.9.a Scrolling Mode

3.9.1. Scrolling Mode to Down

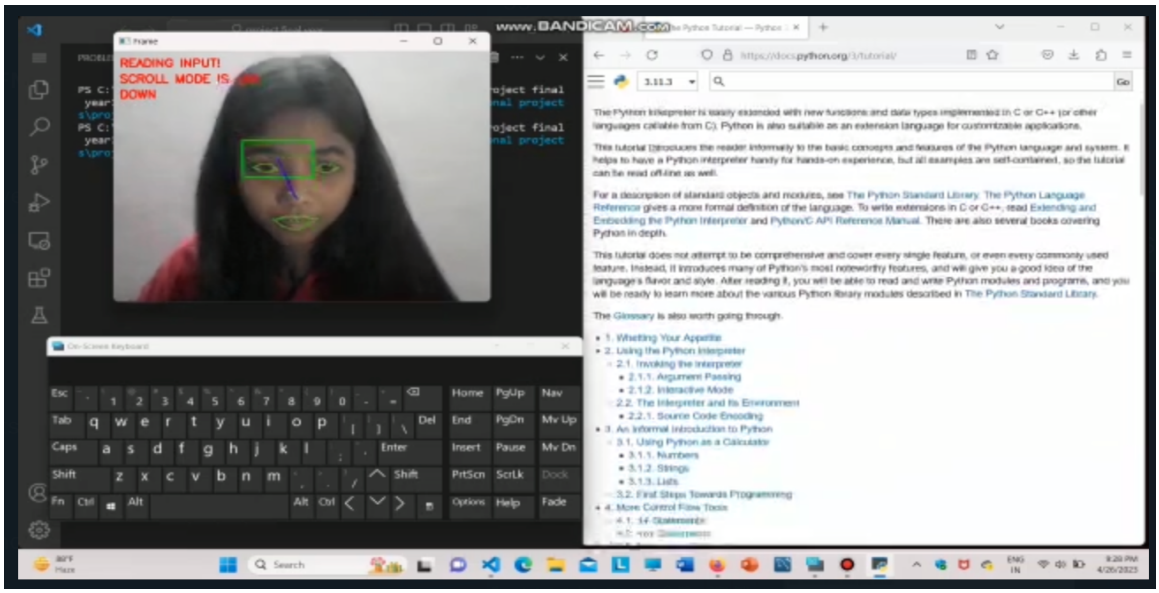

Figure: 3.9.1.a Down Scroll

3.9.2. Scrolling Mode to Up

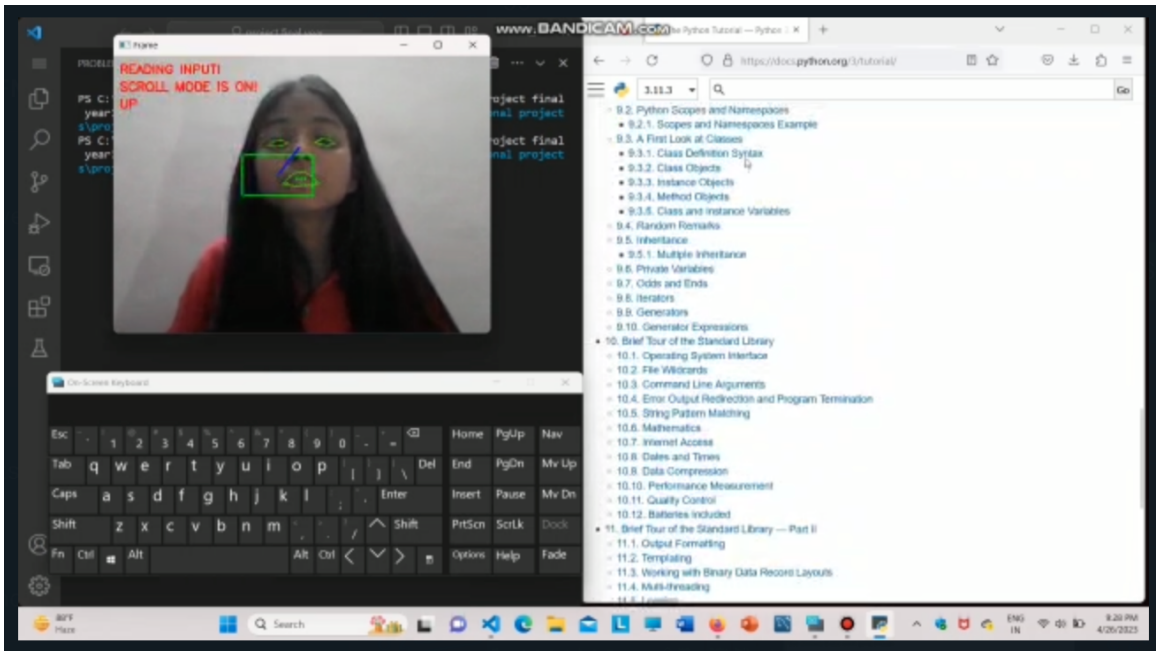

Figure: 3.9.2.a Up Scroll

3.9.3. Scrolling Mode to Right

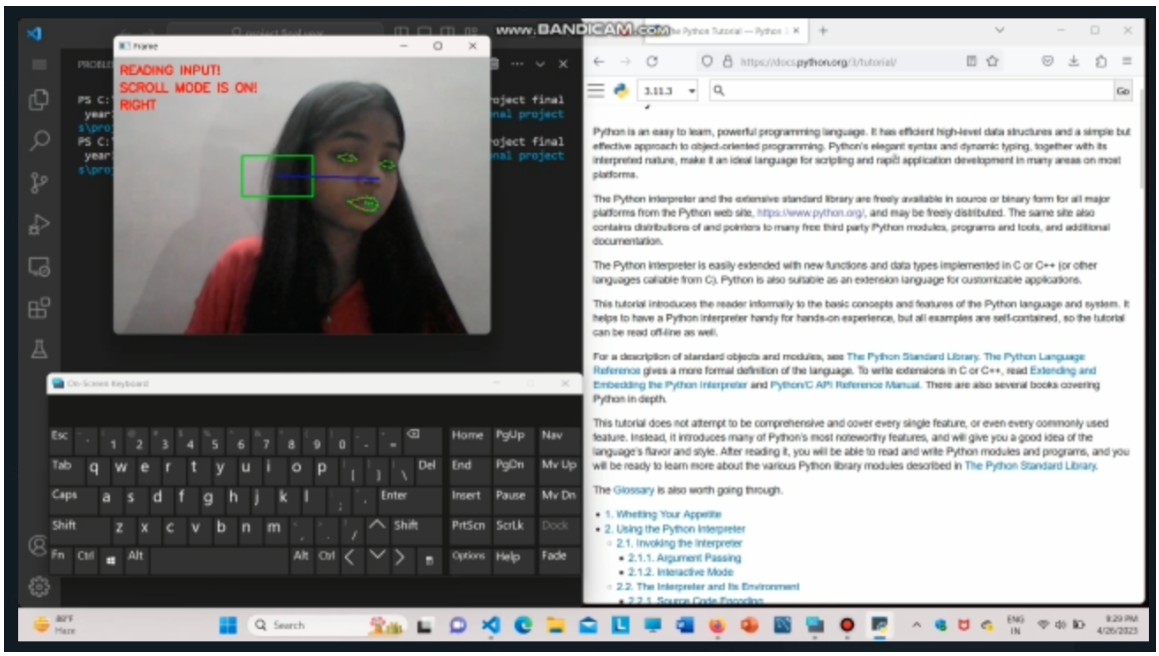

Figure: 3.9.3.a Right Scroll

3.9.4. Scrolling Mode to Left

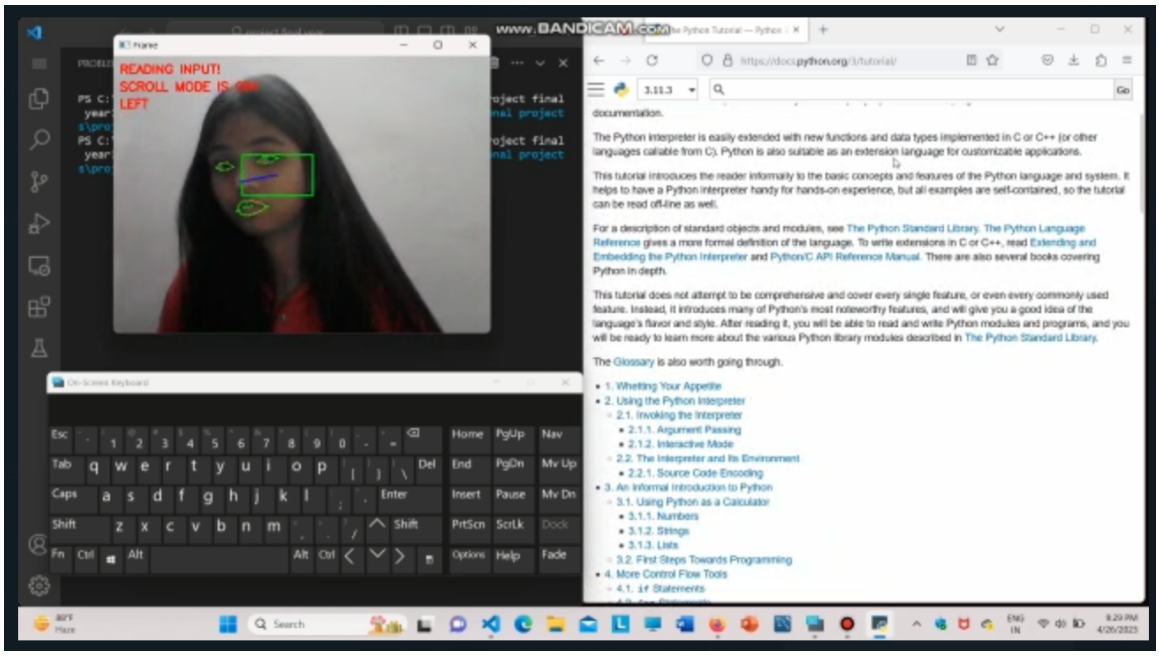

Figure: 3.9.4.a Left Scroll

3.10. Firebase Connection

Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android and web apps Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

3.11. User Interface Details and Prerequisites

The user interface is key to application usability. The application should include content presentation, application navigation, and user assistance. While a comprehensive discussion of effective user interface and web page design is beyond the scope of this document, this section provides some guidelines in the following areas:

1. Content Presentation
2. User Instruction
3. User Navigation
4. Data Manipulation

The user interface must be secure, convenient and extensible. Security requirements include the need to protect authorization information from unauthorized access, the maintenance of customer confidentiality and tracking of payments in progress. The problem of protecting of each party from fraud must also be addressed. The user interface should be efficient in both speed and use. Users have very exacting expectations of any system which involves money. It is essential that such systems provide the user with confidence in their design and implementation. Otherwise a system is likely to encounter overwhelming consumer resistance and fail to gain acceptance. Users must also be confident that they can install configure and use the interface without making unintended purchases. This is especially important on the Internet where users have traditionally tolerated less than perfect security and reliability. A clear and complete description of the payments mechanism may be useful in building user confidence. Complex installation procedures should be avoided.

CHAPTER-4
TESTING AND EVALUATION

4.1. Introduction

Test and Evaluation is the process by which a system or components are compared against requirements and specifications through testing. The results are evaluated to assess progress of design, performance, supportability, etc.Software testing is the process to make sure that the software that you are coding meets client's requirements and it is relatively free from bugs. Software evaluation on the other hand is to identify an already available software which might suit your need.

4.2. System Testing

System testing is the type of software testing that evaluates the overall functionality and performance of a complete and fully integrated software solution. It tests if the system meets the specified requirements and if it is suitable for delivery to the end-users. This type of testing is performed after the integration testing and before the acceptance testing.

4.2.1. Automation testing postman

Tests are automated by creating test suites that can run again and again. Postman can be used to automate many types of tests including unit tests, functional tests, integration tests, end-to-end tests, regression tests, mock tests, etc. Automated testing prevents human error and streamlines testing. Automating testing with your CI/CD Pipeline is easy. Postman allows you to reuse your test suites to create a CI/CD pipeline so you can test at every push. You can seamlessly integrate your Postman instance with Jenkins to create your own CI/CD pipeline or add it to your existing pipeline as a build step.

4.2.2. Load Testing

Load testing is used to identify whether the infrastructure used for hosting the application is sufficient or not. It is used to find the performance of the application is sustainable when it is at its peak of its user load. It tells us how many simultaneous users can the application

handle and the scale of the application required in terms of hardware, network capacity etc.., so that more users could access the application .Its helps to identify the maximum operating capacity of an application as well as any bottlenecks and determine which element is causing degradation.

4.2.3. Postman Testing

Postman is one of the most popular software testing tools which is used for API testing. With the help of this tool, developers can easily create, test, share, and document APIs. This tutorial will help in understanding why Postman is so famous and what makes it unique when compared to other API testing tools. Postman is a standalone software testing API (Application Programming Interface) platform to build, test, design, modify, and document APIs. It is a simple Graphic User Interface for sending and viewing HTTP requests and responses. Application Programming Interface (API) is software that acts as an intermediary for two apps to communicate with each other. We use APIs whenever we use an application like Twitter, Facebook, sending text messages, or checking the weather over the phone.

4.3. Cost Efficient

Economic feasibility analysis is the process of determining whether a new venture is worth the cost and time investment. Economic feasibility analysis is the process of determining whether a new venture is worth the cost and time investment. It is also known as cost benefit analysis.The tools and libraries that we have used like, Visual Studio Code, Numpy, Pandas, data set from Kaggle etc., Learn, all this is open source and easily available to use and learn.So we can say that our project is economically feasible.

CHAPTER-5

TASK ANALYSIS AND SCHEDULE OF ACTIVITIES

5.1 ANALYSIS

1. Study of research papers - (27 June 2022 - 20 July 2022) In this stage we researched different research papers that were available related to the Topic of our Project. We got an idea of what all work has already been done on the selected topic and what we need to do further to make it useful.
2. Perform System Analysis - (21 July 2022 - 1 August 2022) In this step we analyzed the system and got to know about the problems, opportunities and discoveries.
3. Study of Technologies to be used - (2 August 2022 - 15 September 2022) At this stage we were ready to begin with the actual study of the technologies that were to be used in our project. We learnt about Xamarin, C-Sharp, SQL Technologies and their various uses and benefits.
4. Assign Task - (16 October 2022 - 20 October 2022) After learning the technologies and understanding which member will be able to do which type of work, we assigned different parts of the project to all the team members. All the members worked on their part and would also help the other team members if needed.

5.2 DESIGN

1. Perform System Design - (21 October 2022 - 10 November 2022) At this stage, we defined the architecture, product design, modules, interfaces, and data for our project to satisfy the specified requirements.

5.3 BUILD

1. Develop System - (11 November 2022 - 22 February 2023) This was one of the most crucial parts of our project where we built our project and actually started combining the different pieces of the project together to make a final project.

5.4 TESTING

1. Perform Unit Testing - (23 February 2023 - 27 February2023) In unit testing, the individual parts of source code were tested to determine whether they fit to use or not.
2. Perform Integration Testing - (28 February 2023 - 10 March 2023) At this stage, individual modules were combined and tested as a group. It is conducted to evaluate the compliance of a system or component with specific functional requirements.

5.5 MAINTENANCE

1. Deploy System - (11 March 2023 - 31 March 2023) After all the above steps, finally the product was moved from a temporary or development state to a permanent or desired state.

Perform Bug Fixes and Maintenance - (01 April 2023 - 20 April 2023) In this state we used different methods to perform bug fixes and maintenance.

CHAPTER-6
PROJECT MANAGEMENT

6.1 Major risks and contingency plans

Risk Identification forms an important part in Software Development Life Cycle. This gives a view of all the possible risks to the system and whether they can be controlled or not. We may define risks by presenting a questionnaire and answering all the questions relating to it or we also have a table giving risk details and probability of occurrence. The major risk and problem faced are:

1. The model works well only in good lighting on face.
2. Neat facial photo is needed for the model to detect with accuracy.
3. Second issue was to select a angle for feature extraction.
4. We tried various angles including right angle, center , left angle., of which center angle was giving better results.
5. Issues were faced relating to the accuracy of the model we trained in earlier phases which we eventually improved by increasing the input image size and also by improving the dataset.

6.2 Principle learning outcomes

In this report, a facial and eye recognition for user have been developed for physically abled to operate the system without get connected physically to it.We achieved an accuracy of on our dataset. Prediction has been improved after implementing two layers of algorithms in which we verify and predict symbols which are more similar to each other.

CONCLUSION AND FUTURE SCOPE

The eyes gesture recognition is based on fuzzy inference system. The fuzzy inference system is the foundation for the eyes gesture recognition. Eye tracking devices have historically fallen into two categories. The first is passive, and focused on detecting the gaze of the user relative to the rest of the world and in particular what elements of the visible field are currently being focused upon. The second is more active, and considers the eye not as simply a means of observation, but a means of control as well. We think that a device capable of identifying deliberate movements of the eye area (pupils, eyelids and eyebrows), we can provide a new means of interaction that could replace or complement more standard interfaces.

All this come to an conclusion that when a physically disabeled person uses the system so he/she can operate the system by using his/her eyes and facial gestures.

For future aspects we wish to work upon the keyboard and button section without any physical touch as well as you can add your customized shortcuts using facial gestures and eye blinks.

REFERENCES

1. M. Betke, J. Gips and P. Fleming. "The camera mouse: Visual tracking of body features to provide computer access for people with severe disabliites." IEEE Transactions on Neural Systems and Rehabilitation Engineering, 10:1, pages 1-10, March(2002).

2. R.J.K. Jacob and K.S. Karn. “Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises (Section Commentary),” in The Mind's Eye:Cognitive and Applied Aspects of Eye Movement Research, ed. by J. Hyona,R. Radach, and H. Deubel, pp. 573-605, Amsterdam, Elsevier Science (2003).

3. C. Morimoto and M. Mimica. "Eye gaze tracking techniques for interactive applications."Computer Vision and Image Understanding 98", pages 4-24, (2005).

4. Tom M. Mitchell, ―Machine Learning, McGraw-Hill Education (India) Private Limited, (2013).

5. Ethem Alpaydin, ―Introduction to Machine Learning (Adaptive Computation and Machine Learning), The MIT Press (2004).

6. S. Russell and P. Norvig, “Artificial Intelligence: A Modern Approach‖, Prentice Hall, Third Edition, (2009).