B.E. Project Phase-II Report

on

Gesture Based Virtual Keyboard and Mouse

by

Kranti Bhosale (I4122)
Labhesh Chaudhari (I4129)
Prathmesh Chaudhari (I4130)
Mansi Ingale (I4156)

Under the guidance of

Prof. S. A. Nagtilak

Department of Information Technology

Smt. Kashibai Navale College of Engineering, Pune-41

SAVITRIBAI PHULE PUNE UNIVERSITY 2023-2024

Sinhgad Technical Education Society,

Smt. Kashibai Navale College of Engineering, Pune-41

DEPARTMENT OF INFORMATION TECHNOLOGY

Date:

CERTIFICATE

This is to certify that,

KRANTI BHOSALE (I4122)
LABHESH CHAUDHARI (I4129)
PRATHMESH CHAUDHARI (I4130)
MANSI INGALE (I4156)

of class B.E IT have successfully completed their Project Phase-II work on "GESTURE RECOGNITION BASED VIRTUAL MOUSE AND KEYBOARD" at **Smt. Kashibai Navale College of Engineering, Pune** in the partial fulfillment of the Graduate Degree course in B.E at the **Department of Information Technology**, in the academic Year 2023-2024 as prescribed by the Savitribai Phule Pune University, Pune.

Prof.(Mrs).S. A. Nagtilak
Project Guide

Dr. M. L. Bangare
Head of the Department
(Department of Information Technology)

(External Examiner)
Savitribai Phule Pune University, Pune

Dr. A.V. Deshpande
Principal
SKNCOE, Pune

ABSTRACT

Introducing the "Gesture-Based Virtual Mouse and Keyboard," a cutting-edge computer interface technology leveraging computer vision and gesture recognition. It offers eye detection-based virtual mouse control, enabling precise cursor movements through advanced eye-tracking. This eye-controlled mouse provides an alternative input method, particularly beneficial for individuals with physical disabilities.

Additionally, it incorporates hand gesture recognition for virtual keyboard input. Using depth-sensing cameras and machine learning, it interprets natural hand movements to simulate traditional keyboard actions, reducing reliance on physical keyboards.

It aims to achieve user-friendliness, precision, and accessibility by seamlessly integrating eye detection and hand gesture recognition. Users can effortlessly switch between virtual mouse and keyboard functions, enhancing productivity and convenience.

Its success relies on robust computer vision algorithms, real-time processing, and user-friendly interfaces. The technology has diverse applications, from improving accessibility to enhancing overall human-computer interaction efficiency.

The Gesture-Based Virtual Mouse and Keyboard project represents a forward-looking step in technology, making computers more accessible and user-friendly by bridging the gap between users and their devices through natural eye and hand movements. It removes traditional input device barriers, ensuring efficient and enjoyable digital interactions for all.

ACKNOWLEDGEMENT

We are very thankful to all the teachers who have provided us valuable guidance towards the completion of this project work on **Gesture Recognition Based Virtual Mouse and Keyboard.** We express our sincere gratitude towards cooperative department who has provided us with valuable assistance and requirements for the project work.

We are very grateful and want to express our thanks to **Prof. S. A. Nagtilak** for guiding us in right manner, correcting our doubts by giving her time whenever we required, and providing her knowledge and experience in making this project work. We are also thankful to the HOD of our Information Technology department **Dr. M. L. Bangare** for his moral support and motivation which has encouraged us in making this project work. We are also thankful to our Vice Principal **Dr. K. R. Borole** and our Principal **Dr. A.V. Deshpande**, who provided his constant support and motivation that made a significant contribution to the success of this project.

Kranti Bhosale (I4122)
Labhesh Chaudhari (I4129)
Prathmesh Chaudhari (I4130)
Mansi Ingale (I4156)

Table of Contents

List of Tables

List of Figures

Acronyms

AR	Augment reality
CNN	Convolutional Neural Network
HMM	Hidden Markov Model
IMU	Inertial Measurement Unit
KNN	K-Nearest Neighbours
SVM	Support Vector Machine
VR	Virtual Reality

Chapter 1
INTRODUCTION

1.1 Introduction

In an age of fast technological advancement, our relationship with digital gadgets is critical. Although common, the traditional keyboard and mouse combination poses difficulties, particularly for persons with physical limitations. This project aims to address this critical issue by offering a novel solution: the Gesture-Based Virtual Mouse and Keyboard. By utilizing powerful computer vision and gesture recognition technologies, this game-changing technology seeks to improve accessibility and efficiency. It promises to revolutionize how we interact with digital by leveraging natural movements and eye-tracking, making it more accessible and user-friendly for everybody, regardless of physical restrictions.

The touchless virtual keyboard system represents a cutting-edge innovation in human-computer interaction, offering a revolutionary approach to technology text input that transcends traditional keyboard input methods. In today's rapidly evolving technological landscape, where the demand for seamless and intuitive user interfaces continues to rise, the development of touchless input solutions has emerged as a promising frontier in enhancing accessibility, convenience, and hygiene in computing environments. This comprehensive introduction delves deep into the intricacies of touchless virtual keyboards, exploring their evolution, underlying technologies, applications across diverse industries, potential impact on society, and prospects. At the core of the touchless virtual keyboard system lies a fusion of advanced computer vision, machine learning, and human-computer interaction techniques, all meticulously orchestrated to enable users to interact with digital interfaces using natural hand gestures and movements. By harnessing the power of real-time hand tracking and gesture recognition algorithms, coupled with sophisticated eye detection mechanisms, touchless virtual keyboards empower users to navigate and manipulate digital content with unprecedented ease and precision. Gone are the days of cumbersome physical keyboards and cumbersome mouse input; instead, users can seamlessly compose text, navigate menus, and execute commands simply by gesturing in the air, ushering in a new era of touchless computing.

The journey towards the realization of touchless virtual keyboards has been characterized by a convergence of technological advancements and interdisciplinary collaboration. From the early experiments in gesture-based computing to the advent of sophisticated deep learning models for hand and eye tracking, each milestone has contributed to the evolution of touchless input solutions. Today, with the advent of high-resolution webcams, powerful processors, and robust software frameworks, touchless virtual keyboards have transitioned from experimental prototypes to practical tools with tangible applications across a myriad of domains. In the realm of accessibility, touchless virtual keyboards hold immense promise for individuals with physical disabilities, providing them with a lifeline to digital communication and interaction. By removing the physical barriers associated with traditional input devices, such as keyboards and mice, touchless virtual keyboards empower users with mobility impairments to express themselves, communicate with others, and access digital resources with unparalleled independence and dignity. Moreover, for individuals with conditions such as arthritis or carpal tunnel syndrome, which may limit their ability to use conventional input devices, touchless virtual keyboards offer a welcome respite, alleviating discomfort and facilitating seamless interaction with digital devices.

Beyond accessibility, touchless virtual keyboards have found a fertile ground for innovation and experimentation in a myriad of industries and sectors. In the realm of healthcare, for instance, touchless virtual keyboards have emerged as invaluable tools for healthcare professionals, enabling them to input patient data, access medical records, and control diagnostic equipment without the need for physical contact, thereby minimizing the risk of cross-contamination and infection transmission. Similarly, in the field of education, touchless virtual keyboards have revolutionized the learning experience, allowing students to participate in interactive lessons, collaborate on projects, and engage with educational content in an immersive and intuitive manner. Moreover, in the realm of entertainment and gaming, touchless virtual keyboards have opened up new vistas of creativity and interactivity, enabling gamers to control characters, navigate virtual environments, and execute complex manoeuvres using nothing but their hands and gestures. From virtual reality (VR) experiences that transport users to fantastical realms to augmented reality (AR) applications that blur the lines between the digital and physical worlds, touchless virtual keyboards have become indispensable tools for game developers and enthusiasts alike, offering unparalleled immersion and engagement.

1.2 Aim and Motivation

The major goal of this project is to develop a Gesture-Based Virtual Mouse and Keyboard system that uses eye recognition for mouse control and hand gestures for keyboard input. In our system, we want to achieve user-friendliness, accuracy, and accessibility. Our specific objectives include the development of strong eye-tracking technology for mouse control, the implementation of efficient hand gesture detection for keyboard input, and the seamless integration of these features.

1.3 Background and Need of Project

For decades, traditional input devices such as mice and keyboards have been the industry standard for interacting with digital devices. However, these technologies pose significant challenges, particularly for persons with physical limitations. People with restricted hand movement, for example, may find it difficult, if not impossible, to utilize a typical mouse and keyboard. Furthermore, the repetitive and frequently unnatural movements required by standard input devices can cause discomfort and agony over time.

Computer vision and gesture recognition advances have enabled the development of more natural and accessible user interfaces. Gesture recognition systems are capable of tracking and interpreting human body motions, allowing users to operate computers and other digital devices without the necessity of conventional input devices.

There are various reasons for the need for gesture recognition-based virtual mouse and keyboard systems:

- To make computers and digital gadgets more accessible to those with physical disabilities.

- To alleviate the strain and pain caused by standard input devices.
- To develop a more natural and efficient manner of interacting with computers and digital devices.
- To make technology more accessible and inclusive to all individuals.

1.4 Key Objectives

1. Develop Precise Eye-Tracking Technology:
   - This objective focuses on creating eye-tracking technology that accurately detects and tracks the movement of the user's eyes.
   - Precision is crucial here to ensure that the technology can capture subtle eye movements and translate them into meaningful interactions.
   - The goal is to develop a system that can reliably interpret where a user is looking on a screen or within a physical environment.
2. Implement Accurate Hand Gesture Recognition:
   - This objective involves creating a system that can accurately recognize and interpret hand gestures made by the user.
   - Accurate recognition ensures that the system can understand the intended gestures reliably, enabling seamless interaction.
   - The technology should be capable of identifying a wide range of gestures with precision to provide users with intuitive control over the system.
3. Ensure Seamless Integration of Functionalities:
   - This objective aims to integrate both eye-tracking and hand gesture recognition functionalities seamlessly into a single system.
   - Integration involves ensuring that the two technologies complement each other effectively, allowing users to switch between eye-tracking and gesture control fluidly.
   - The system should provide a cohesive user experience, where both functionalities work together harmoniously to enhance usability and accessibility.
4. This objective emphasizes Enhance Accessibility for Disabled Users:
   - Making the technology accessible to users with disabilities, such as mobility impairments.

- The system should be designed to accommodate various accessibility needs, allowing users with disabilities to interact with digital interfaces effectively.
- Features like customizable gestures, voice commands, and adaptive interfaces can enhance accessibility for a diverse range of users.

5. Prioritize User-Friendly Design:
   - This objective focuses on designing the system with the user's experience in mind.
   - User-friendly design principles should guide the development process to ensure that the technology is intuitive and easy to use.
   - Clear interfaces, straightforward controls, and intuitive feedback mechanisms contribute to a positive user experience.
6. Achieve Precision and Accuracy:
   - Precision and accuracy are essential for both eye-tracking and hand gesture recognition functionalities.
   - The system should be finely tuned to minimize errors and accurately interpret user inputs.
   - Calibration processes and advanced algorithms can help achieve the desired level of precision and accuracy.
7. Optimize for Real-Time Performance:
   - Real-time performance is critical to ensure responsive interactions between the user and the system.
   - Optimization efforts should focus on reducing latency and processing times to deliver instantaneous feedback to user inputs.
   - Efficient algorithms and hardware acceleration techniques can help achieve real-time performance.
8. Provide Versatile Interaction Options:
   - This objective involves offering users a variety of interaction options to suit their preferences and needs.
   - Alongside eye-tracking and hand gesture recognition, the system could support additional input methods such as voice commands or touch controls.

1.5 Organization of the Report

Chapter 1: Introduction - In this chapter, the introduction includes the Problem Definition, Justification of the Problem, the need for a new system, and Updating the previous system.

Chapter 2: Literature Survey - It synthesizes related research done on this topic. Literature survey includes different research done methods using their algorithms and which area the research lacks along with that it also includes existing system architecture and its detailed explanation and working.

Chapter 3: Project Statement - This chapter includes Background information, desired future state current state, Differences between the actual and intended state and the effect of the problem.

Chapter 4: System Requirements - This chapter gives overall tools and techniques used in the system and the information related to them.

Chapter 5: System Design – This chapter contains architectural information on both the proposed system and the contributions made to it, as well as the existing ones.

Chapter 6: System Implementation- This phase involves the actual development and integration of the virtual mouse and keyboard system, including coding, hardware setup, and software configuration. It transforms design concepts and theoretical models into a functional, operational system ready for use and testing.

Chapter 7: GUI Working Modules and Experimental Results - These modules represent the graphical user interface components that allow users to interact with the virtual mouse and keyboard system. They facilitate user inputs, visual feedback, and the overall usability of the system. Results validate the system's functionality, accuracy, and responsiveness under various conditions and user interactions.

Chapter 8: Testing - Testing involves systematically evaluating the system to identify and fix bugs, ensure reliability, and verify that it meets specified requirements. It is crucial for ensuring that the virtual mouse and keyboard system operates as intended in real-world scenarios.

Chapter 9: Conclusion - The conclusion summarizes the overall findings, highlights the success and limitations of the project, and may suggest areas for future improvements or research. It encapsulates the final assessment of the system's viability and performance.

Chapter 2
LITERATURE SURVEY

2.1 Literature Survey

Table 2.1 Literature Survey

Sr No	Ref. Name & Publish Year	Idea or Methodology	Metric and Advantages	Problem Found
1.	Operating Virtual Keyboard using Gesture Recognition and Machine Learning. [IJARSCT - 2023]	-Idea is to implement the haarcascade algorithm by using OpenCV and python [Numpy, Scipy,PyAutoGUI] to detect the hand motion. -It uses straight forward rule classifier. -To print the output character it uses live captures and CNN. -Frame is converted to image that extract position of hand over simulated keyboard.	-Laptop's webcam is the sole piece of HW -Use of Haarcascade which is maintained in OpenCV library. -PyautoGUI is used to automate the tasks -Accuracy is up to 97%	-Keyboard is useful for only fore & middle fingers. -Vulnerable to ambient noise, privacy problem to being overhead.
2	Non-touch Character Input System Based on Hand Tapping Gesture Using Kinect Sensors. [IEEE 2020]	It uses the Hidden Markov Model along with Haarcascade algorithm, which is mainly developed for Japanese script Hiragana. -Kinect cannot provide the precise outline so that the system can clearly distinguish between stretched fingers close to each other.	Average accuracy is about 94.3% -Provides extra features like delete, add space and character set switch -Accuracy is different for P and S hand	-Prize of Kinect Sensor is very High. -Keyboard is designed for 8 hiraganas at one time
3.	Hand Gesture Recognition Based Text Input Method for AR/VR Wearable Device [IEEE 2020]	-Idea is to develop the system with wearable devices like TypingRing, LEAP motion sensors.	-100% training, 96.12% validation accuracy, 2% validation loss with experimental configuration on 150 epoch with predefined	-High HW prices -Very high system specifications

		-It uses SVN and HMM. -3D finger CAPE to detect the clicked activities.	steps to achieve this accuracy rates.	-Difficult to detect fingertips in real time application
4.	Towards a Virtual Keyboard Scheme based on Wearing One Motion Sensor Ring on Each Hand [IEEE 2020]	-The smart ring integrates a 6-DoF Inertial Measurement Unit (IMU) and a 3-DoF magnetometer sensor for collecting motion data during typing. -New keyboard layout is designed, by changing the previous, rectangular layout to an arc structure,	-Running times of k-NN and SVM classifiers are 12.0536ms and 884.32ms, respectively	-Complex calculations -Higher dimensional feature description -Cost of HW
5.	A Virtual Keyboard Implementation Based on Finger Recognition [IEEE-2017]	The virtual keyboard is customized and printed on plain paper so that it can be put on any oblique plane or stuck on a wall. The device camera is then used for key recognition based on hand skin tone and fingertip location	Virtual keyboard can be printed on a plain paper or projected on a wall or other flat surface. This makes the virtual keyboard environmentally friendly because of its minimal resource use.	Small number of users used in this feasibility study
6.	Gesture Recognition based Virtual Mouse and Keyboard [IJRASET-2023]	The implemented system enables the control of mouse cursor movement by tracking the movement of the user's eyeballs and hand gestures. The system replaces the conventional input devices such as a mouse and keyboard by combining their functionalities	Assistive technology has also played a role in preventing the spread of infectious diseases. With touchless interfaces and voice-activated controls, assistive devices can reduce the need for physical contact and limit the spread of germs	Not able to recognize more subtle hand movements or incorporate haptic feedback to provide a more realistic and intuitive user experience
7.	Real-time virtual mouse system using RGB-D images and fingertip detection [ieeexplore-2020]	The fingertip location is mapped to RGB images to control the mouse cursor based on a virtual screen. The system tracks fingertips in real-time at 30 FPS on a desktop computer using a single CPU and Kinect V2	The proposed method overcomes the limitations of most current virtual-mouse systems. It has many advantages, e.g., working well in changing light levels or with complex backgrounds, accurate fingertip tracking at a longer distance, and fingertip tracking of multiple people.	This study still suffers from several limitations that are mainly inherited from Microsoft Kinect. Therefore, our next work aims to overcome those limitations and improve the Fingertip tracking

				intend to expand our system to handle more gestures and interact with other smart environments
8.	Virtual Mouse And Keyboard For Computer Interaction [IJTMS-2022]	Method used for Hand gesture recognition is based on Convolutional Neural Networks (CNNs).The computer can be controlled virtually and can perform left click, right click, scrolling functions ,volume control, brightness control and other computer cursor function without the use of a physical mouse	Does not require any training phase for gesture recognition and the training accuracy of the system is 96.37%	The system performance is very well in good lighting conditions only

Samuel Solomon [1] Human face detection has been a challenging issue in the areas of image processing and patter recognition. A new human face detection algorithm by primitive Haar-Cascade algorithm combined with three additional weak classifiers is proposed in this paper. The three weak classifiers are based on skin hue histogram matching, eyes detection and mouth detection. First, images of people are processed by a primitive Haar-Cascade classifier, nearly without wrong human face rejection (very low rate of false negative) but with some wrong acceptance (false positive). Secondly, to get rid of these wrongly accepted non-human faces, a weak classifier based on face skin hue histogram matching is applied and a majority of non-human faces are removed. Next, another weak classifier based on eyes detection is appended and some residual non-human faces are determined and rejected. Finally, a mouth detection operation is utilized to the remaining non-human faces and the false positive rate is further decreased. With the help of OpenCV, test results on images of people under different occlusions and illuminations and some degree of orientations and rotations, in both training set and test set show that the proposed algorithm is effective and achieves state-of-the-art performance. Furthermore, it is efficient because of its easiness and simplicity of implementation.

J. Shin and C. M. Kim, [2] stated that There have been a lot of studies on the text input system using the image-based hand gesture recognition. However, hand gesture languages such as sign languages, finger alphabets, and aerial handwriting treated in the previous works have some problems to be commonly used. The aerial handwriting requires much time for writing and recognition. The sign languages and finger alphabets demand quite a knowledge and practice for using it, which results in restricting the number of their users. As a solution to the problems, this paper proposes a new character input system based on hand tapping gestures for Japanese hiragana and English characters that can be used to facilitate human-computer interaction. The hand tapping gestures are motions for tapping keys on aerial virtual keypads by hands, which can be effectively used as a hand alphabet by anyone, including hearing impaired individuals. For hiragana characters, the hand used for tapping a key and the number of stretched fingers of the hand decide the consonant part of characters, and thereby the aerial virtual keypad. The character to be entered is determined by tapping the key on the virtual keypad corresponding to the desired vowel. Because we adopt a key layout similar to the Japanese and English flick keyboard of smart phones, our hand tapping gestures can be easily used by anyone with only a brief description. The users can effectively interact with computers by using our non-touch input system where only the Kinect sensor is used without any keyboard, mouse or body-worn device. We expect that our character input system will open a new channel for human-computer interaction.

L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua stated that [3] Hand gestures, whether static or dynamic, are a field of intense study and have several potential uses for human-computer interaction in real-time systems. Static and dynamic hand gestures are rudimentary ways for human-computer interaction. This paper presents a technique for the text input method which is hand-gesture-recognition based. This compact hand-based text input system is proposed for augmented reality (AR) and virtual reality (VR) devices. To recognize and classify hand gestures, the hand image is captured by a standard camera. After, the hand is segmented using background subtraction, and then the segmented hand gesture is input in the trained neural network for gesture recognition. Finally, hand movements are tracked and recorded using a convex hull algorithm. The corresponding written character is passed to a trained neural network. The proposed architecture is tested and the experimental results are compared with other methods, which showed that the proposed method performed better than traditional methods and achieved 96.12% accuracy, achieved accuracy is overall better than existing methods.

In this paper, [4] we present an improved ring-type virtual keyboard scheme that can achieve impressive performance with only one smart ring on a finger of each hand. The smart ring integrates a 6-DoF Inertial Measurement Unit (IMU) and a 3-DoF magnetometer sensor for collecting motion data during typing. First, a new keyboard layout is designed, by changing the previous rectangular layout to an arc structure, this method increases the difference in attitude angle between adjacent keys, which greatly improved the keystroke recognition accuracy. Secondly, other than the attitude angle feature, we also adopt acceleration data, gyroscope data and magnetometer data to describe the subtle differences between different keystrokes motion. Then, feature importance evaluation and feature correlation analysis were used to select features with high contribution rate and low similarity to describe keystrokes. Finally, nine effective features were selected from the attitude angle and magnetometer data for the final keystroke recognition. By weighing the number of selected features, recognition speed and recognition accuracy of training models, the keystroke recognition speed can increase by nearly 4 times while ensuring 98.53% of the keystroke recognition accuracy. This new ring-type virtual keyboard input scheme has the advantages in portability, small volume, and lower cost over many existing human-computer interface methods.

Y. Zhang, W. Yan and A. Narayanan [5] stated that, they develop a new type of virtual keyboard that allows users to type on any plane to any device. The virtual keyboard is customized and printed on plain paper so that it can be put on any oblique plane or stuck on a wall. The device camera is then used for key recognition based on hand skin tone and fingertip location. If the fingertip remains on a key for a predetermined amount of time, the program will regard this key as an input. That experiments show how various customized virtual keyboards allow users to input text of their choice if they have a real keyboard in front of them and with no identifiable performance compromise. The overall recognition rate (true positives divided by all samples) of all inputs is 94.62%. The best average input recognition rate is at 97.7% from keyboard (a) under natural lighting condition, and the worst case is 90.7% from keyboard (b) under lamplight.

Pratiksha Kadam, Prof. Minal Junagre, Sakshi Khalate, Vaishnavi Jadhav, Pragati Shewalep[6] stated that , This research project aims to use computer vision to develop an optical mouse and keyboard that can be operated through hand movements. The computer camera will capture images of different hand gestures made by the user, and the mouse pointer or cursor on the computer screen will move accordingly. Different hand gestures can be used to execute right and left-clicks. Similarly, the keyboard functions can be performed using different hand actions, such as using a finger to select an alphabet and a four-digit swipe left or right. The virtual mouse and keyboard can be used wirelessly or externally, and the only hardware required for the project is a webcam. The implemented system enables the control of mouse cursor movement by tracking the movement of the user's eyeballs and hand gestures. The system replaces the conventional input devices such as a mouse and keyboard by combining their functionalities. The main aim of this system is to provide a comfortable data entry method that is versatile and portable, especially for small mobile devices. The virtual mouse and keyboard system utilize gesture recognition, cognition, and image processing to move the mouse cursor in accordance with the eyeball movement and to perform keyboard functions using hand gestures.

Dinh-Son Tran & Ngoc-Huynh Ho & Hyung-Jeong Yang1,Guee Sang Lee [7] stated that , Using fingertip tracking as a virtual mouse is a popular method of interacting with computers without a mouse device. In this work, we propose a novel virtual mouse method using RGB-D images and fingertip detection. The hand region of interest and the center of the palm are first extracted using in-depth skeleton-joint information images from a Microsoft Kinect Sensor version 2, and then converted into a binary image. Then, the contours of the hands are extracted and described by a border-tracing algorithm. The K cosine algorithm is used to detect the fingertip location, based on the hand-contour coordinates. Finally, the fingertip location is mapped to RGB images to control the mouse cursor based on a virtual screen. The system tracks fingertips in real-time at 30 FPS on a desktop computer using a single CPU and Kinect V2. The experimental results showed a high accuracy level; the system can work well in real-world environments with a single CPU. For the group of two, the fingertip detection accuracy was highest with 93.25%. The worst case was the group of six with an accuracy of 53.35%. The accuracies of the three, four, and five groups were 89.78%, 78.03%, and 65.38%, respectively. The results show that the accuracy decreases as the number of people in the group increases.

Mishaha MK, Manjusha MS[8] stated that , In the proposed virtual mouse and keyboard system, this limitation can be overcome by employing webcam or a built-in camera for capturing of hand gestures and coloured object. Method used for Hand gesture recognition is based on Convolutional Neural Networks (CNNs).The computer can be controlled virtually and can perform left click, right click, scrolling functions, volume control, brightness control and other computer cursor function without the use of a physical mouse Hand Gesture Recognition. Python is a system that can detect the gesture of hand in a real time video. Python OpenCV library can be used to capture gestures from computer's internal camera or web cam. In the proposed system, the Object tracking method has been used to track the colored objects that help to operate on this system using the laptop webcam. By using the Object tracking system, the keyboard and its basic operations space, enter, backspace etc. The system performance is very well in good lighting conditions and also this overcome the problem with the background that is gesture can recognize with any background. Nevertheless the system is little bit fast responsive as compared to the other system which have been developed earlier as it does not require any training phase for gesture recognition and the training accuracy of the system is 96.37.

2.2 Gap Analysis

The goal is to improve computer interaction by replacing traditional mouse and keyboard inputs with eye tracking and hand gestures. The pursuit of accuracy and user-friendliness in this cutting-edge system, guided by prior research in the subject, is at the heart of our problem. We acknowledge that the significant gap resides in achieving the maximum levels of accuracy when utilizing hand gestures and Haar Cascade for keyboard and mouse operation, a lesson learned from earlier investigations. These results highlight the need to enhance the system's capability to correctly understand and react to user gestures, whether they involve hand movement or gaze tracking. By utilizing the knowledge from prior research. We seek to close these gaps in order to provide a richer user experience and expand the potential of our ground-breaking system.

2.3 Problem Definition

Improving the accuracy and user-friendliness of computer interface is the current problem. Our goal is to create a novel touchless input system employing eye tracking and hand gestures as an alternative to conventional mouse and keyboard inputs, which are frequently constrained. How to make sure that this system correctly understands and reacts to user gestures both hand movements for keyboard input and gaze tracking for mouse like functions is the main challenge. Our project's major objective is to close the user-friendliness gap between accuracy and calibration while preserving accuracy.

Chapter 3
SYSTEM REQUIREMENT AND SPECIFICATIONS

3.1 Hardware requirements:

Processor: Intel core 5

Ram size: 8GB

Hard disk capacity: 500GB

Monitor type: 15 Inch shading screen

Keyboard type: web console

Camera: Logitech G20

3.2 Software requirements:

Operating System: Windows 10

IDE: SPYDER

Programming language: Python

Documentation: MS-Off

Chapter 4
SYSTEM ARCHITECTURE

4.1 Existing System Architecture

Our methodology uses a multifaceted strategy to accomplish the objectives outlined in our problem statement. We develop a cutting-edge touchless input method by utilizing cutting-edge technologies like eye tracking and hand motion recognition. For keyboard input, hand gestures are used, and the Haar Cascade approach helps recognize eye movements for mouse-like control. Eye tracking technology is used to simultaneously record and examine the user's gaze. These interconnected parts function together to improve our system's precision and usability. As part of our technique, we rigorously optimize the code to guarantee responsiveness and real-time performance. We develop our touchless input solution through interdisciplinary collaboration that combines computer vision and human-computer interaction principles.

4.2 Proposed System Architecture

Capture Input: The system begins by collecting user inputs such as hand motions and eye tracking data.

1. Processing of Hand Gestures: The first phase is hand gesture recognition, in which the system evaluates and interprets the user's hand movements. This entails analysing hand motions' location, direction, and pattern.
2. Eye Tracking: Concurrently, the program uses eye tracking technologies to identify where the user's sight is on the screen.
3. Gesture Integration: The algorithm combines data from hand gesture recognition and eye tracking, allowing it to make more educated judgements.
4. Recognition of Gestures: The Gesture Recognition System then processes the combined data, identifying particular user instructions or control signals. This stage focuses on differentiating between motions and transferring them to appropriate actions.

5. User Interface Improvements: When the algorithm recognizes the user's intent, it executes the appropriate commands to update the user interface. These changes are either shown on the screen or delivered as commands to apps or the operating system.
6. Real-time Feedback: The system delivers real-time feedback to ensure that the user's input is correctly executed, hence improving the overall user experience.

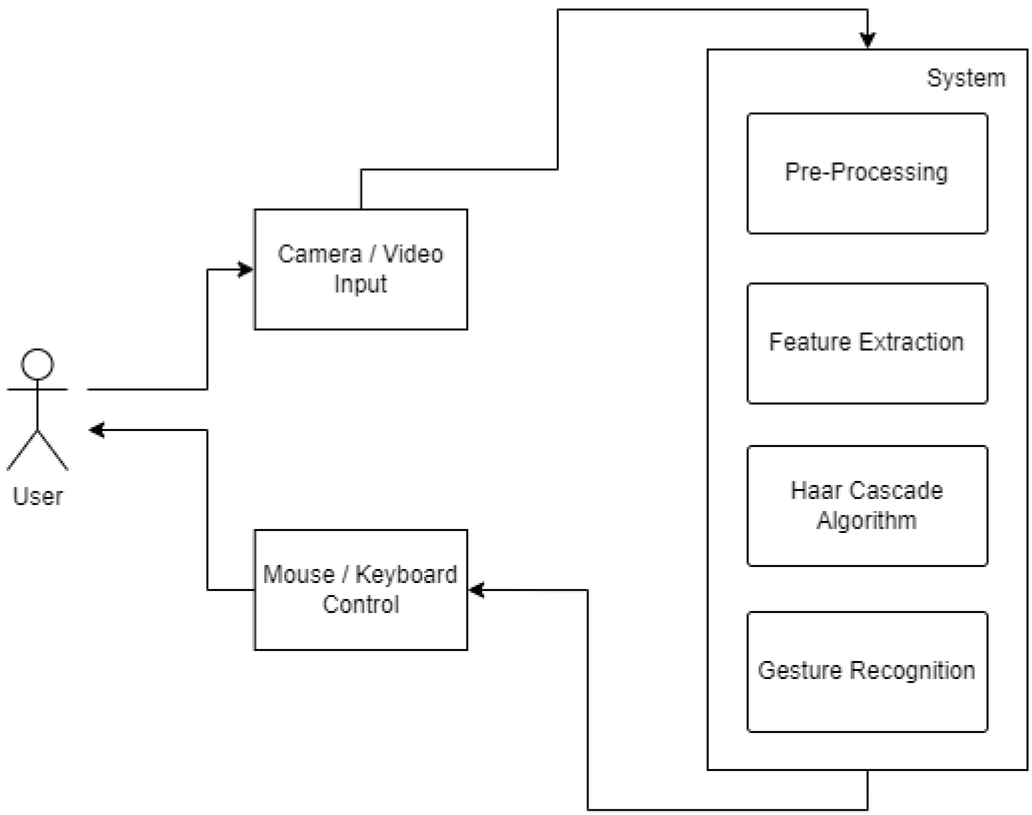


Fig. 5.1 System Architecture

A system diagram for a Gesture-Based Virtual Keyboard and Mouse typically outlines the various components and their interactions to illustrate how the system works. This system combines hand or body gestures to input text or control a computer's cursor, mimicking the functions of a physical keyboard and mouse. Above is an explanation of the key components in such a diagram.

User: This is the individual using the Gesture-Based Virtual Keyboard and Mouse. The user's hand or body movements are tracked and interpreted to generate input commands.

Gesture Input Device: This device captures the user's gestures. It is a camera-based system.

Gesture Recognition Software: The captured gesture data is processed by gesture recognition software. This software interprets the movements and converts them into meaningful commands. It employs algorithms like computer vision, machine learning, or sensor fusion to recognize gestures accurately.

Gesture Recognition Software: The captured gesture data is processed by gesture recognition software. This software interprets the movements and converts them into meaningful commands. It employs algorithms like computer vision, machine learning, or sensor fusion to recognize gestures accurately.

Feedback Display: A visual or audio feedback display is often included to give the user information about the gestures being recognized. This can help users adjust their gestures for accuracy.

4.3 Novelty

The novelty of the project lies in its innovative approach to combining eye-tracking and hand gesture recognition for virtual mouse control and keyboard input. By integrating two distinct interaction modalities into a single system, the project enhances accessibility and efficiency for users with diverse needs and abilities. This seamless transition between input modes provides flexibility and reduces cognitive load on the user, resulting in a more intuitive and natural interaction experience. Furthermore, the project opens up possibilities for future applications in areas such as gaming, virtual reality, and human-computer interaction, where multimodal interaction plays a crucial role in creating immersive user experiences.

Chapter 5
HIGH LEVEL SYSTEM DESIGN

5.1 Use-Case Diagram:

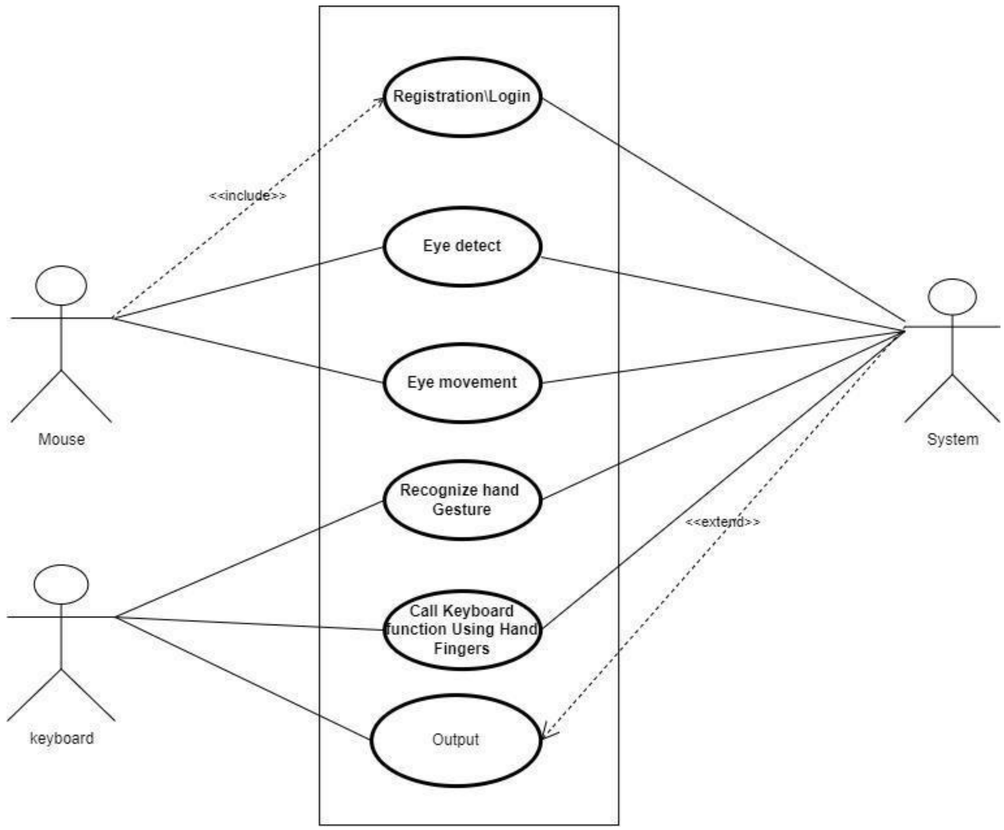


Fig. 5.1 Use-Case Diagram

i. Actors:

User: The primary actor interacting with the system. This include Keyboard and Mouse

System: The software system that handles the user's requests.

ii. Use-Case

For Mouse Controller

a. **User Registration:** allows the user to create a new account by providing necessary information (e.g., username, email, and password).

b. **User Login:** Enables registered users to log in to their accounts using their credentials.

c. **Eye Detection:** The system detects the user's eyes and tracks their movements for interaction purposes.

d. **Eye Movement Tracking:** Monitors and interprets the user's eye movements for various functions within the system.

For Keyboard Controller

a. **Recognize Hand Gestures:** Identifies and interprets hand gestures made by the user for controlling the system.

b. **Call Keyboard Function Using Hand Finger:** Allows the user to invoke keyboard functions (e.g., typing, special commands) by using hand and finger gestures.

iii. **Output:** Represents the system's response to the user's inputs, including the execution of keyboard functions and other relevant actions.

5.2 Sequence Diagram:

It demonstrates how user gestures are interpreted and translated into corresponding mouse and keyboard actions by the system, allowing the user to control these input devices through gestures. The diagram captures the dynamic nature of these interactions over time.

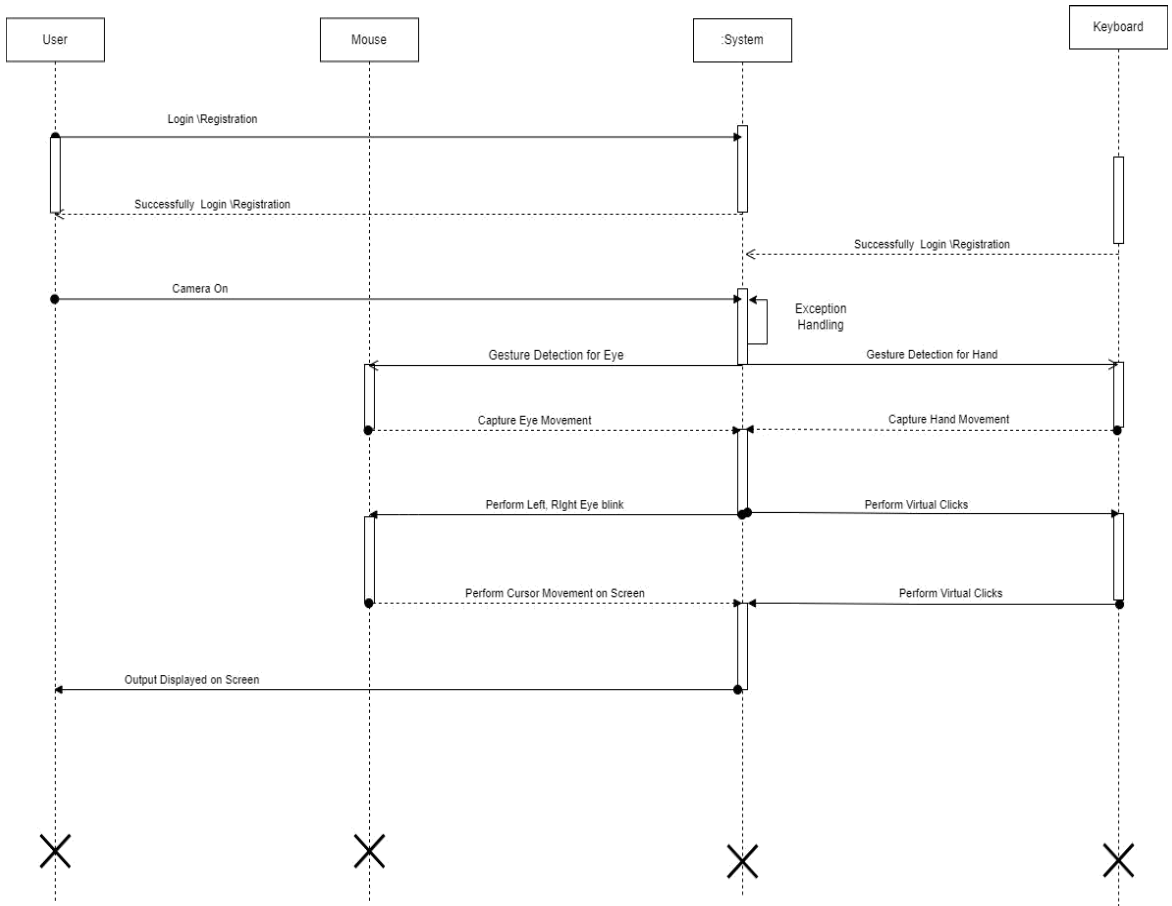


Fig. 5.2 Sequence Diagram

User initiates a gesture to interact with the system. This gesture could be a hand movement, finger swipe, or any other motion that represents a specific action, such as moving the mouse cursor or typing a key.

System is the first component to receive the gesture input from the user. The System component interprets the gesture and converts it into corresponding mouse movements or Keyboard clicks. It's responsible for processing the gesture and determining whether it's intended for controlling the mouse or keyboard.

Keyboard: The Keyboard component is responsible for simulating keypresses on the virtual keyboard. It generates keyboard events corresponding to the keys that the user intends to type based on their gestures.

Mouse: Mouse is the first component to receive the gesture input from the user. The Mouse component interprets the gesture and converts it into corresponding mouse movements or clicks.

5.3 Activity Diagram:

Start: This is the initial point of the activity diagram, representing the beginning of the process.

User performs a gesture: This step signifies that the user begins by making a gesture, which can be recognized by the system. Gestures could include hand movements, finger motions, or any other form of input that the system is designed to interpret.

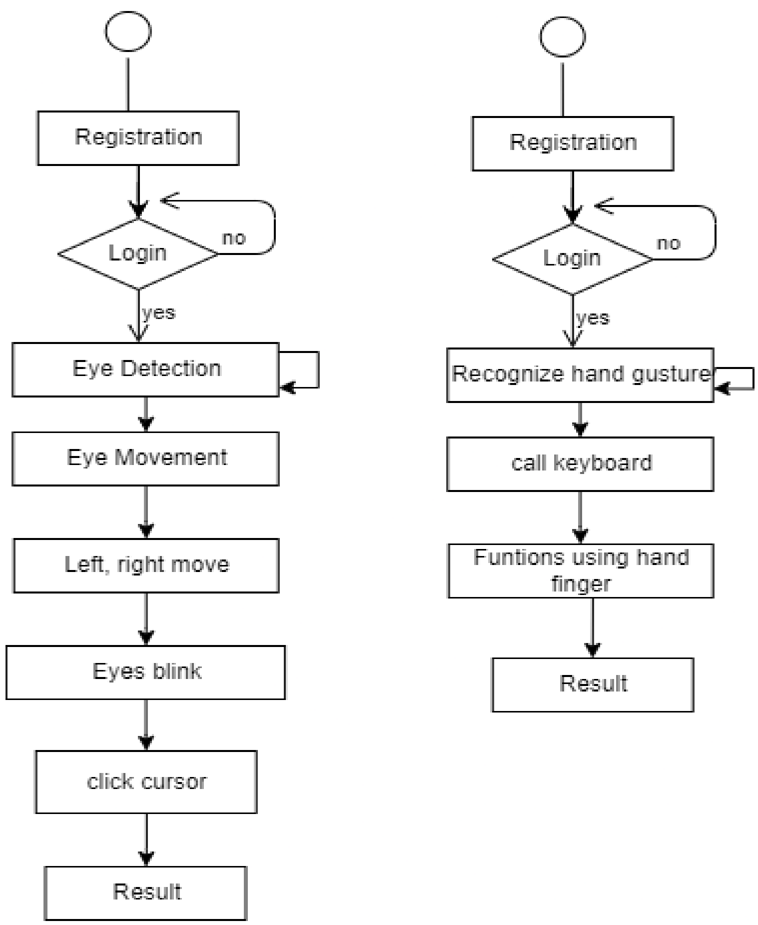


Fig.5. 3 Activity Diagram

System recognizes the gesture: Once the user performs a gesture, the system's gesture recognition module processes it to understand the user's intent. This could involve computer vision or other sensor technologies to interpret the gesture.

i. For Mouse:

Gesture is for mouse control: If the user's initial gesture is not for typing but for mouse control, the system enters the mouse control mode. System enters mouse control mode: In mouse control mode, the system switches to interpreting gestures as commands for controlling the mouse pointer.

User performs mouse gestures (e.g., move, click, and scroll): The user performs gestures that correspond to mouse actions, such as moving the mouse cursor, clicking, or scrolling.

System translates gestures into mouse actions: The system interprets the user's mouse gestures and translates them into corresponding mouse actions, like moving the cursor on the screen or simulating mouse clicks.

System moves the cursor or performs click/scroll actions: Based on the translated gestures, the system moves the mouse cursor on the screen or executes actions like clicking on items or scrolling.

ii. For Keyboard:

If gesture is for typing: The system decides whether the user's gesture is intended for typing. If it is, the process continues in the typing mode.

System displays a virtual keyboard: In typing mode, the system displays a virtual keyboard on the screen. This virtual keyboard can be controlled and manipulated through gestures.

User selects a key on the virtual keyboard: The user selects a specific key on the virtual keyboard by performing the appropriate gesture over the desired key.

System registers the selected key: The system recognizes the key that the user has chosen and registers this input.

System displays the selected key on the screen: The selected key is displayed on the screen, typically in a text field or wherever the user's input is intended.

5.4 Class Diagram:

User represents the user of the system and may have attributes like **userID** and **username**. **Gesture** represents a gesture performed by the user. It has attributes like **gestureID** and **type**, where **GestureType** could be an enumeration defining different types of gestures.

VirtualKeyboard represents the virtual keyboard component of the system. It has a layout attribute that defines the keyboard layout and methods for displaying and interacting with the keyboard.

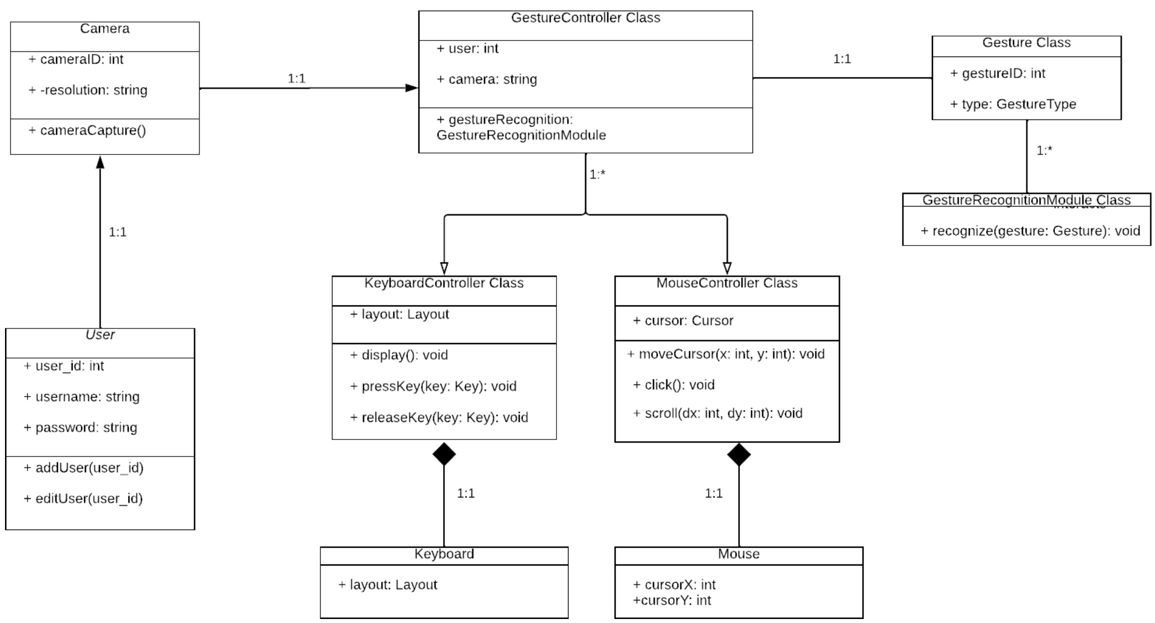


Fig.5.4 Class Diagram

MouseControl represents the mouse control component. It has a cursor attribute to track the cursor's position and methods for moving the cursor, clicking, and scrolling.

The relationships are as follows:

User has a 1:1 relationship with **Camera**, indicating that each user is associated with one camera.

User has a 1:1 relationship with **Gesture Controller** and **Mouse Controller**, indicating that each user has one controller for gestures and one for the mouse.

Gesture Controller and **Mouse Controller** have a 1:1 relationship with **GestureRecognitionModule** and **Mouse**, respectively, indicating that each controller manages one input device.

GestureController has a 1:* relationship with Gesture, indicating that it can manage multiple gestures.

MouseController has a 1:1 relationship with **Mouse**, indicating that it manages a single virtual mouse.

KeyboardController has a 1:1 relationship with **Keyboard**, indicating that the keyboard is managed by the keyboard controller.

5.5 DFD (Data Flow Diagram):

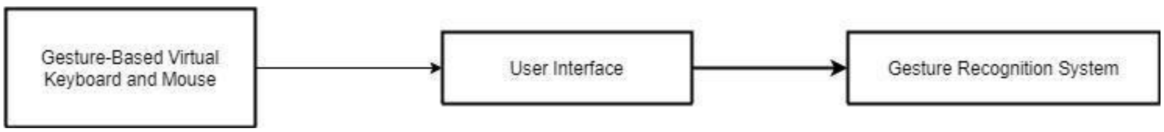


Fig. 5.5.1 DFD Level 0

At Level 0, we have a high-level view of the system, showing the main processes and external entities. In this case, the main processes are the "**User**," the "**Gesture Recognition System**," and the "**Virtual Keyboard and Mouse System**."

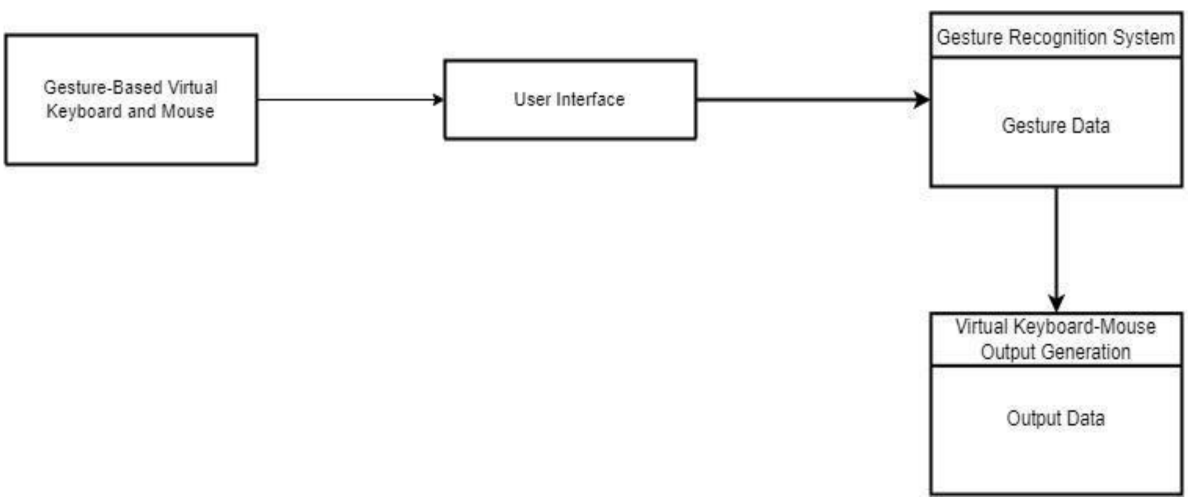


Fig. 5.5.2 DFD Level 1

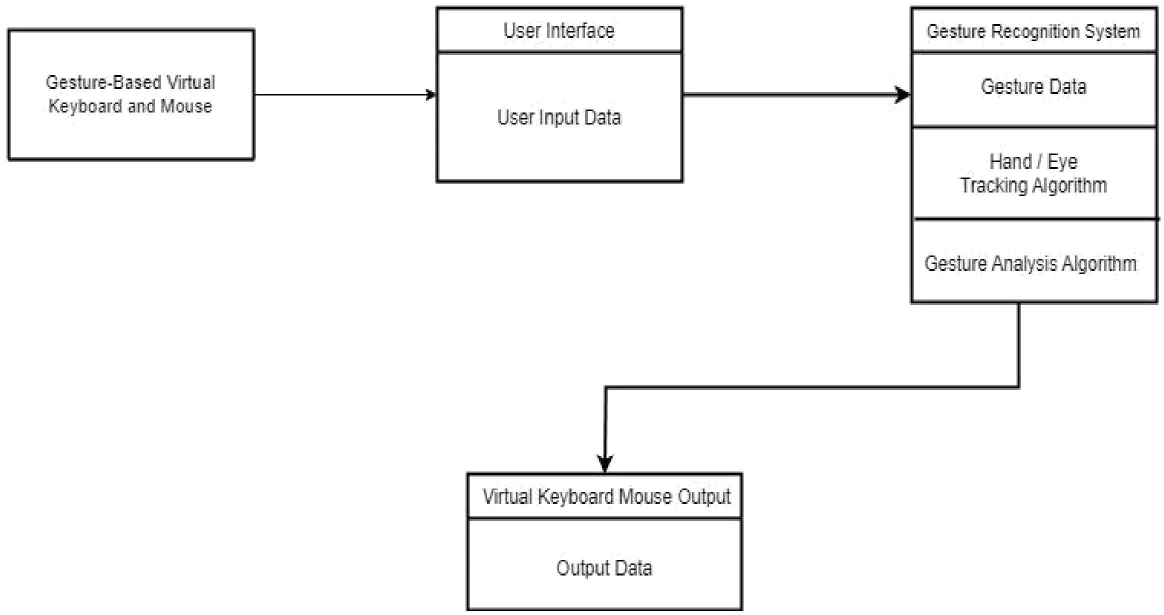


Fig. 5.5.3 DFD Level 2

In this Level 2 DFD, we've broken down the main processes further to include various subsystems responsible for different tasks within the Gesture-Based Virtual Keyboard and Mouse System. Here are some key details:

i. **User Interface**: Receives user input data such as touch gestures, voice commands, and hand movements. Sends this data to the Gesture Recognition Subsystem.

ii. **Gesture Recognition System**: Processes the input data to recognize gestures. Contains two key components: Gesture Processing Subsystem and Hand Tracking Algorithm. Sends gesture data to the Gesture Analysis Algorithm.

iii. **Tracking Algorithm**: Tracks the movement of the user's hand or and Eye Movement through input devices.

iv. **Virtual Keyboard and Mouse Output Generation System**: Generates virtual keyboard inputs (e.g., key presses) and mouse inputs (e.g., cursor movements). Sends these inputs to the user's system for interaction.

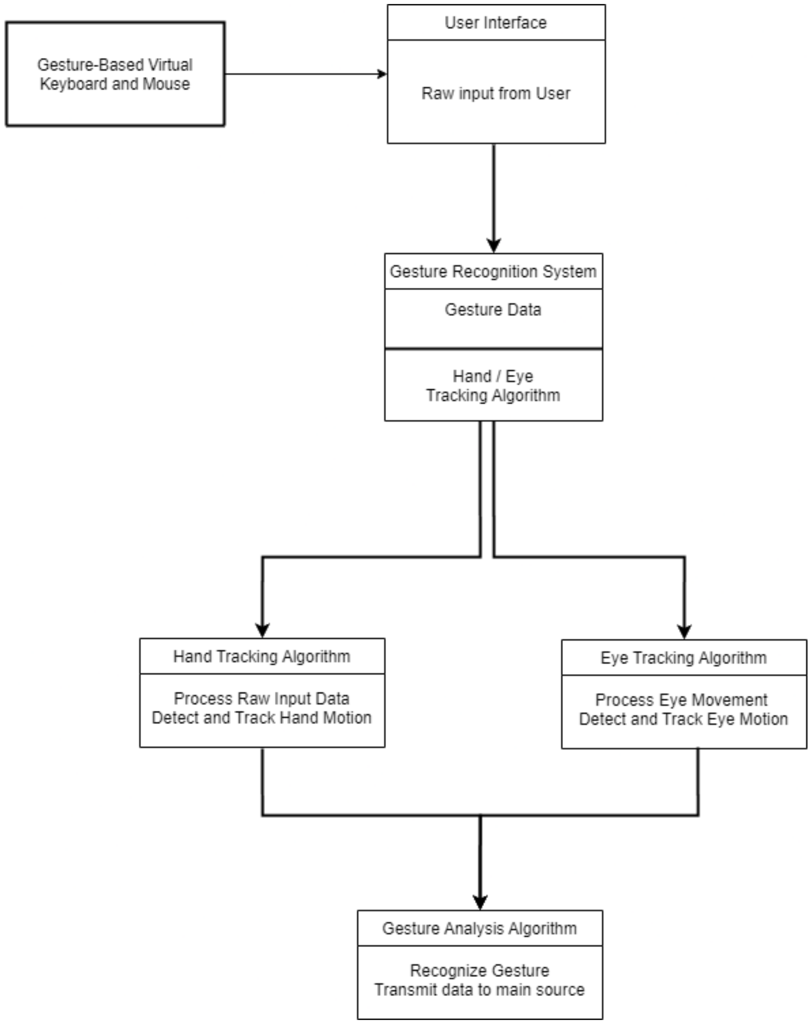


Fig. 5.5.4 DFD Level 3

1. User Interface (Process):

   - Input: User gestures (from hand and eye tracking).

   - Output: Processed data for command execution.

2. Hand Tracking (Process):

   - Input: User's hand gestures.

   - Output: Extracted hand gesture data.

3. Eye Tracking (Process):

   - Input: User's eye movements.

   - Output: Eye tracking data.

4. Gesture Recognition System (Process):

   - Input: Hand gesture data and eye tracking data.

- Output: Recognized commands or mouse control data.

5. User Interface (UI) Output (Process):
   - Input: Recognized commands or mouse control data.
   - Output: UI updates and commands for applications.

5.6 Package Diagram

1. Gesture-based Virtual Keyboard and Mouse (Top-Level Package):

This is the highest-level package and represents the entire system. It encapsulates all the sub-packages and elements related to the Gesture-based Virtual Keyboard and Mouse system.

2. User Interface Package:

The User Interface package represents the part of the system responsible for interaction with the user through gestures. This package contain various components, classes, and interfaces related to the user interface, including gesture recognition, touch input, and the virtual keyboard/mouse interface.

3. System Controller Package:

The System Controller package is responsible for managing the overall functionality and control of the Gesture-based Virtual Keyboard and Mouse system. It include components for processing user gestures, interpreting gestures as commands, and coordinating the communication between the user interface and the hardware components.

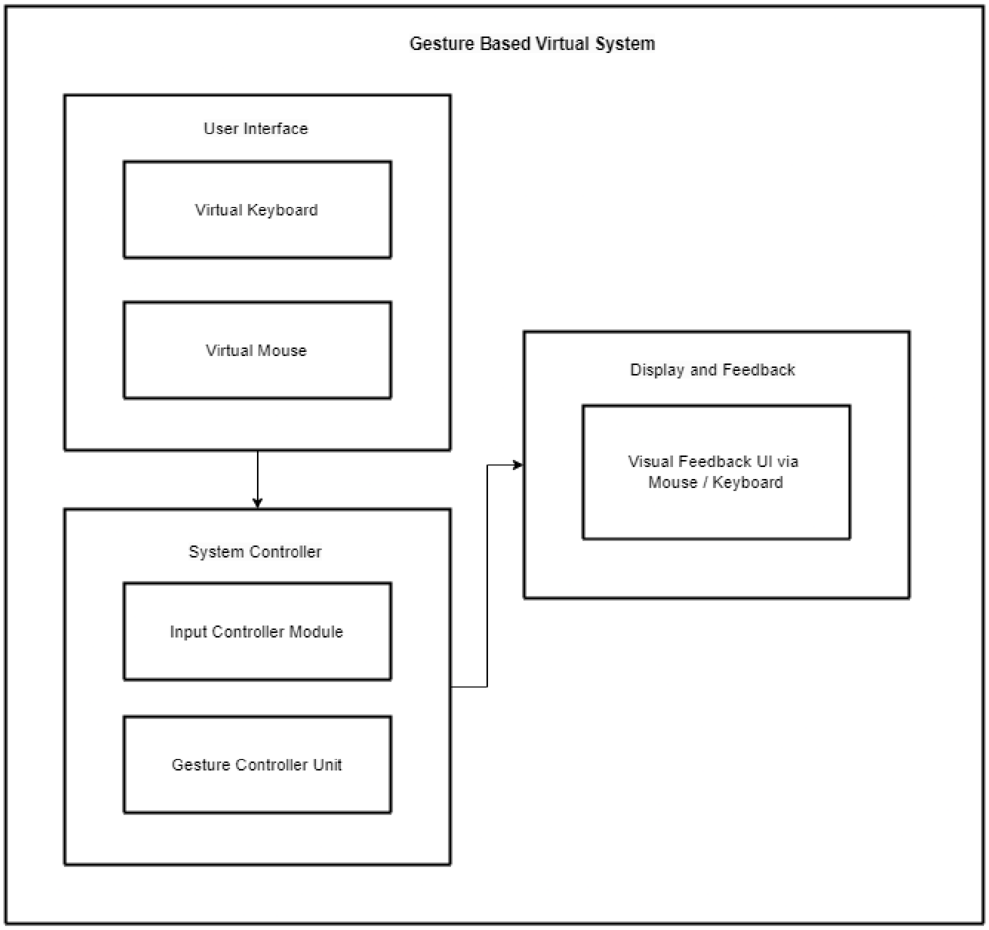


Fig. 5.6 Package Diagram

4. Display / Feedback Package:

The Display / Feedback package manages the visuals and feedback given to the user. It may include components or modules responsible for displaying the virtual keyboard on the screen and providing feedback to the user in response to their gestures. This feedback can include visual cues, sound feedback, or haptic feedback through devices like vibration motors.

Chapter 6
SYSTEM IMPLEMENTATION

6.1 Assumptions

The touchless virtual keyboard and mouse system operate under a set of fundamental assumptions that shape its design, functionality, and user experience. First and foremost, it is assumed that users possess devices equipped with suitable hardware capabilities, notably a webcam or camera capable of capturing video feed for hand and eye detection. This foundational hardware element enables the system to track users' hand movements and gestures accurately, facilitating intuitive interaction with the virtual keyboard and mouse interfaces. Additionally, a stable internet connection is presumed, supporting various system functionalities such as accessing online resources or transmitting user inputs to remote servers if necessary. Operating system compatibility is another critical consideration, with the system designed to seamlessly integrate with commonly used platforms like Windows, macOS, and Linux, ensuring broad accessibility and usability across different devices.

Furthermore, the system assumes that users are familiar with gesture-based input methods and comfortable utilizing hand gestures to interact with digital interfaces. This familiarity enhances user adoption and acceptance of the touchless input system, promoting a seamless transition from traditional input methods to touchless alternatives. In terms of interface design, the system relies on a predefined keyboard layout, encompassing letters, numbers, symbols, and special characters commonly found in conventional keyboards. This standardized layout ensures consistency and ease of use for users accustomed to traditional keyboard configurations. Moreover, the system prioritizes optimized user interfaces characterized by intuitive navigation and clear visual feedback, enhancing user engagement and satisfaction. Accurate hand and eye detection capabilities are fundamental assumptions underlying the system's functionality, enabling precise selection of letters and cursor control without significant errors or delays. Additionally, the touchless nature of the system alleviates concerns regarding hygiene, particularly in environments where cleanliness is paramount, such as healthcare facilities or public spaces. By eliminating physical contact with input devices, the system contributes to improved hygiene practices, safeguarding user health and well-being. Privacy and security considerations are also integral to the system's design, with robust data encryption protocols and adherence to privacy regulations ensuring the protection of sensitive user information captured during hand and eye detection. Finally, the system's scalability and customization capabilities enable integration with diverse applications and environments, accommodating varying user preferences and requirements. These assumptions collectively form the foundation of the touchless virtual keyboard and mouse system, guiding its development and implementation to deliver a seamless and intuitive user experience across different contexts and use cases.

In addition to the existing assumptions outlined, several key considerations further contribute to the touchless virtual keyboard and mouse system's functionality and effectiveness. Firstly, it is assumed that the system's hand and eye detection algorithms are optimized to adapt to varying environmental conditions, including changes in lighting, background clutter, and user distance from the camera. These optimizations ensure reliable performance across different settings and scenarios, enhancing the system's versatility and usability. Secondly, the system assumes seamless integration with existing computer hardware and software configurations, including compatibility with popular operating systems such as Windows, macOS, and Linux. This compatibility ensures widespread adoption and ease of implementation for users across diverse computing environments. Furthermore, the touchless virtual keyboard and mouse system are designed with security and privacy considerations in mind, adhering to industry-standard encryption protocols and data protection measures to safeguard user information and interactions. It is assumed that the system implements robust authentication mechanisms to verify user identity and prevent unauthorized access, thereby ensuring user privacy and confidentiality. Additionally, the system is assumed to offer customizable settings and preferences, allowing users to personalize their interaction experience based on individual needs and preferences. This customization capability enhances user satisfaction and engagement, fostering a more tailored and user-centric computing environment. Overall, these assumptions contribute to the touchless virtual keyboard and mouse system's effectiveness, reliability, and user-friendliness, positioning it as a cutting-edge solution for enhanced computer interaction in various contexts and applications.

6.2 Algorithm

Step 1: Initialization

1.1 Initialize the system.

1.2 Load necessary libraries and dependencies.

Step 2: User Authentication

2.1 Prompt the user to authenticate.

2.2 Verify user credentials.

Step 3: Main Menu Display

3.1 Present the user with a main menu.

3.2 Display options for Virtual Keyboard and Virtual Mouse.

Step 4: User Input Selection

4.1 Wait for user input.

4.2 If Virtual Keyboard selected, proceed to Step 5.

4.3 If Virtual Mouse selected, proceed to Step 9.

Step 5: Virtual Keyboard Operation

5.1 Render virtual keyboard interface.

5.2 Activate webcam for hand gesture detection.

Step 6: Hand Gesture Recognition

6.1 Continuously analyze video feed.

6.2 Detect and track hand movements.

6.3 Recognize gestures for letter selection.

Step 7: Letter Selection Process

7.1 Map gestures to virtual keyboard keys.

7.2 Highlight selected keys for user feedback.

Step 8: Text Display and Utilization

8.1 Display selected letters in real-time.

8.2 Allow text input and editing actions.

Step 9: Virtual Mouse Operation

9.1 Render virtual mouse interface.

9.2 Activate webcam for hand gesture detection.

Step 10: Cursor Control and Interaction

10.1 Track hand movements for cursor control.

10.2 Recognize gestures for mouse actions.

Step 11: End of Interaction

11.1 Continue monitoring gestures until mode switch or exit.

11.2 Provide options for mode switching or system exit.

Step 12: Shutdown and Cleanup

12.1 Gracefully terminate the system.

12.2 Release allocated resources.

Step 13. End Algorithm

This algorithm outlines the step-by-step process involved in the operation of the touchless virtual keyboard and mouse system, incorporating user authentication, input mode selection, gesture recognition, and interactive feedback mechanisms to deliver a seamless and intuitive user experience.

6.3 Flowchart

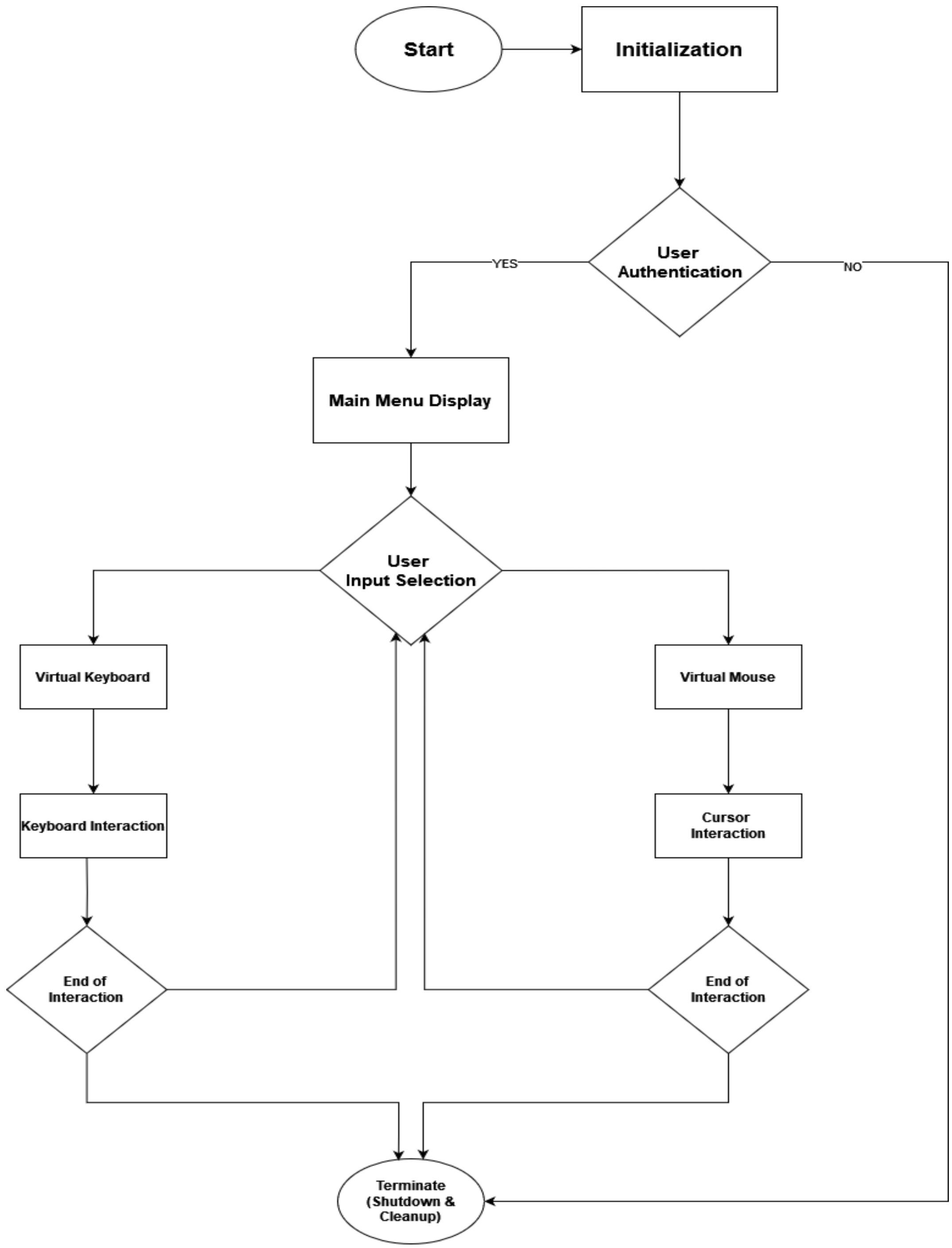


Fig. 6.3 Flowchart Virtual Mouse and Keyboard System

6.4 Methodologies

The touchless virtual keyboard and mouse system leverage several key methodologies to enable its functionality. Firstly, computer vision plays a pivotal role in hand and eye detection. This methodology involves the use of advanced algorithms and techniques to analyze video feed from the webcam or camera in real-time. By identifying and tracking the user's hand movements and gestures, the system accurately interprets hand gestures as commands for letter selection on the virtual keyboard or cursor movement on the virtual mouse interface. Additionally, computer vision techniques are employed to recognize specific gestures associated with mouse actions, such as clicking or dragging, enhancing the system's versatility and usability. Furthermore, human-computer interaction (HCI) principles guide the design and implementation of the system's user interface. HCI methodologies focus on optimizing the interaction between humans and computers to ensure a seamless and intuitive user experience. In the context of the touchless virtual keyboard and mouse system, HCI principles dictate the layout and functionality of the virtual keyboard and mouse interfaces, emphasizing clarity, consistency, and ease of use. By adhering to HCI best practices, the system ensures that users can interact with the virtual keyboard and mouse interfaces effectively, regardless of their level of technical expertise.

Moreover, machine learning algorithms contribute to the system's accuracy and robustness in hand and eye detection. Through the process of training on large datasets of hand and eye images, machine learning models can learn to recognize and differentiate between various hand gestures and eye movements with high precision. This allows the system to adapt to different users' preferences and environments, improving its performance over time through continuous learning and refinement. Additionally, the system incorporates principles of accessibility and inclusivity, aiming to cater to users with diverse needs and abilities. Accessibility methodologies focus on designing interfaces and interactions that are accessible to individuals with disabilities or impairments, ensuring equal access to digital resources and functionalities. By providing alternative input methods such as touchless interaction through hand gestures, the system promotes inclusivity and empowers users with physical limitations to navigate and interact with digital content effectively. Overall, the touchless virtual keyboard and mouse system integrate computer vision, HCI, machine learning, and accessibility methodologies to deliver a seamless and intuitive user experience.

A) Haar cascade algorithm for eye detection:

Paul Viola and Michael Jones presented an efficient object detection technique in their paper "Rapid Object Detection using a Boosted Cascade of Simple Features" (2001) that uses Haar feature-based cascade classifiers [2]. This method is based on machine learning, and it involves training a cascade function with a large number of both positive and negative images. Next, it's applied to identify objects in additional pictures.

To train the classifier, the algorithm first requires a large number of positive images, or images with faces, and negative images, or images without faces. After that, we must extract its features, Haar features, as seen in the image. They resemble the convolutional kernel exactly. Each feature is a single value obtained by subtracting the sum of pixels under the white rectangle from the sum of pixels under the black rectangle.

Currently, each kernel's potential sizes and locations are utilized to compute a large number of features—roughly 16000 features. We must ascertain the total of the pixels beneath the white and black rectangles for every feature computation. They introduced the integral image as a solution. However, the majority of these features—of all the ones we calculated—are unimportant. The Adaboost algorithm is used to solve this problem. It eliminates up to 6000 features. We do this by applying every feature to every training image. It determines the optimal threshold for each feature to categorize the faces as positive or negative. According to the paper, 95% accuracy in detection is achieved with just 200 features. They used about 6000 features in their final setup. So now you take an image. Take each 24x24 window. Apply 6000 features to it. Check if it is face or not.

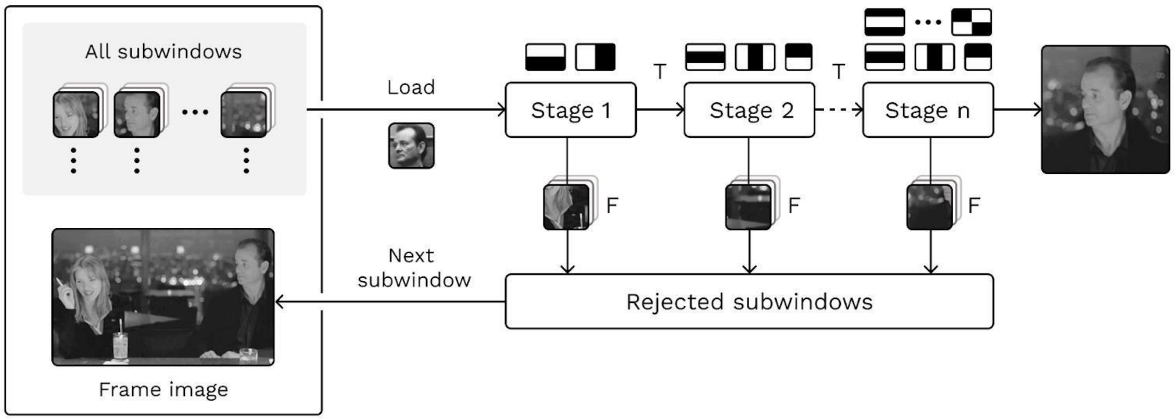


Fig. 6.4.1. Haar Cascade Algorithm Working

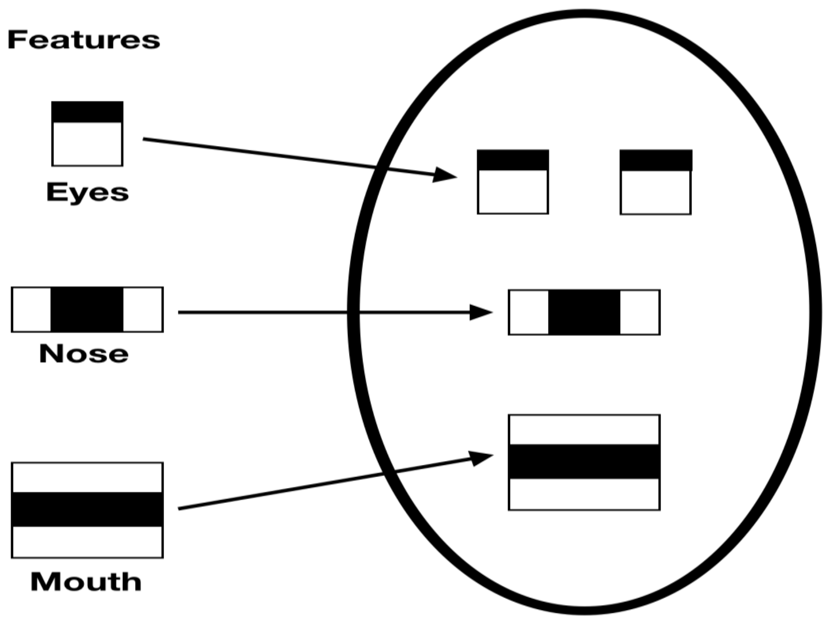


Fig. 6.4.2 Eyes Extraction Model

B) Hand gesture recognition for hand detection

Techniques for recognizing hand gestures usually entails multiple crucial phases. First, information is obtained using cameras or depth sensors that record hand movements in pictures or videos. After that, preprocessing methods such as segmentation to isolate the hand and noise reduction are used to improve the quality of the data. Then, pertinent features that include both temporal and spatial aspects of hand movements are extracted from the preprocessed data. These characteristics accurately depict various hand gestures.

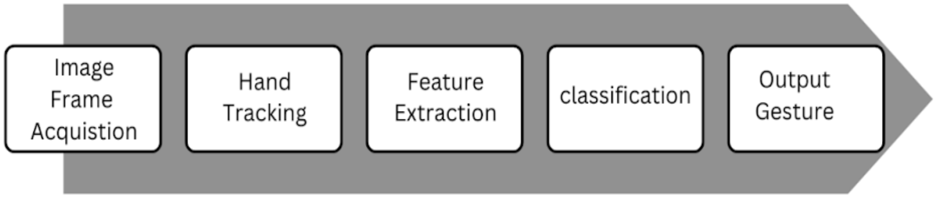


Fig. 6.4.3 Hand Gesture Recognition

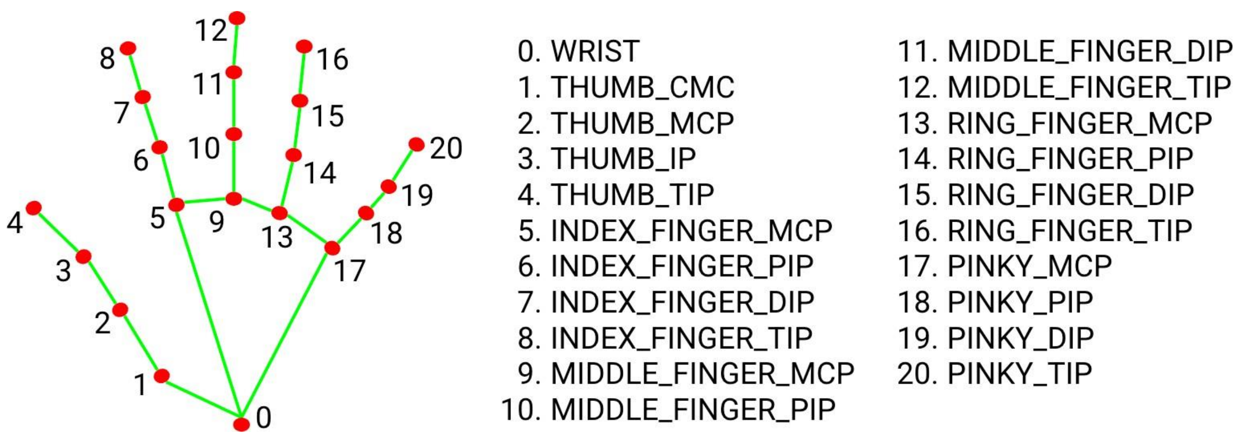


Fig. 6.4.4 Hand and Fingers Detection

Chapter 7
GUI / EXPERIMENTAL RESULTS

7.1 GUI / Working Modules

1) Home Page

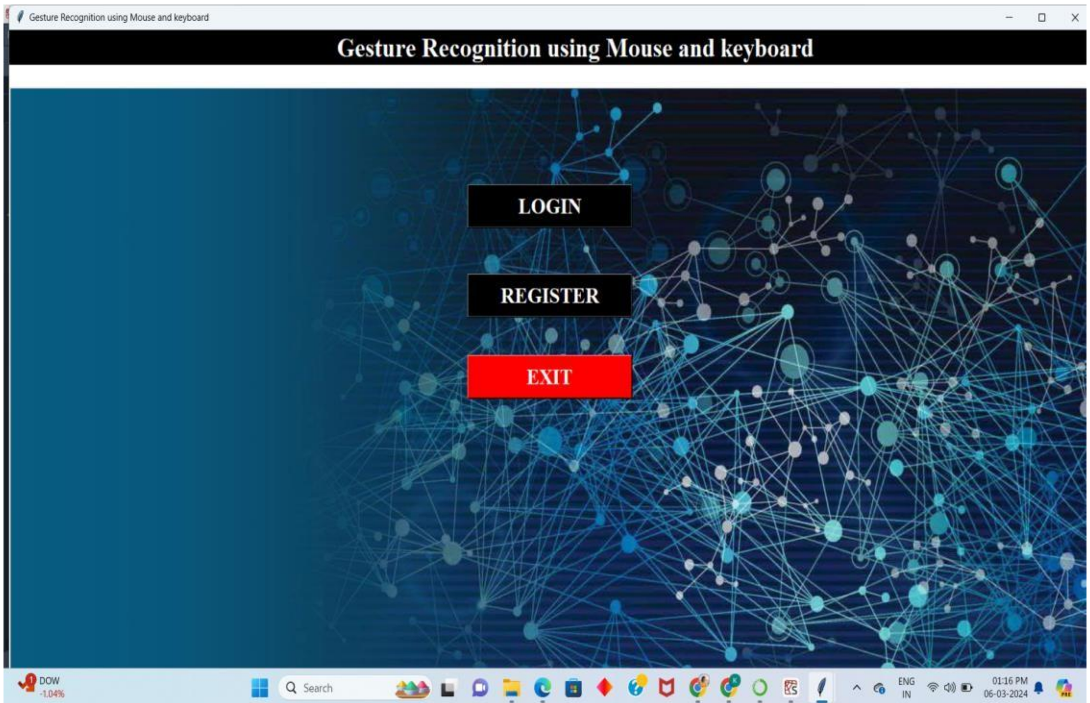


Fig. 7.1.1 Home Page

2) Registration Page

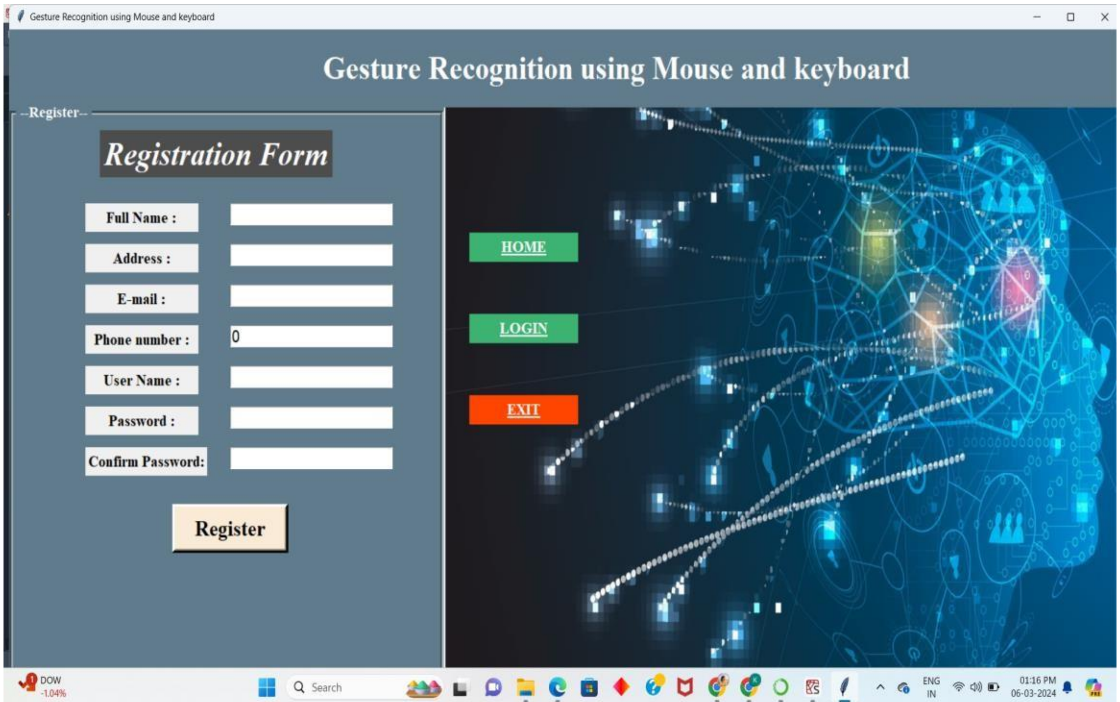


Fig. 7.1.2 Registration Form

3) Login Page

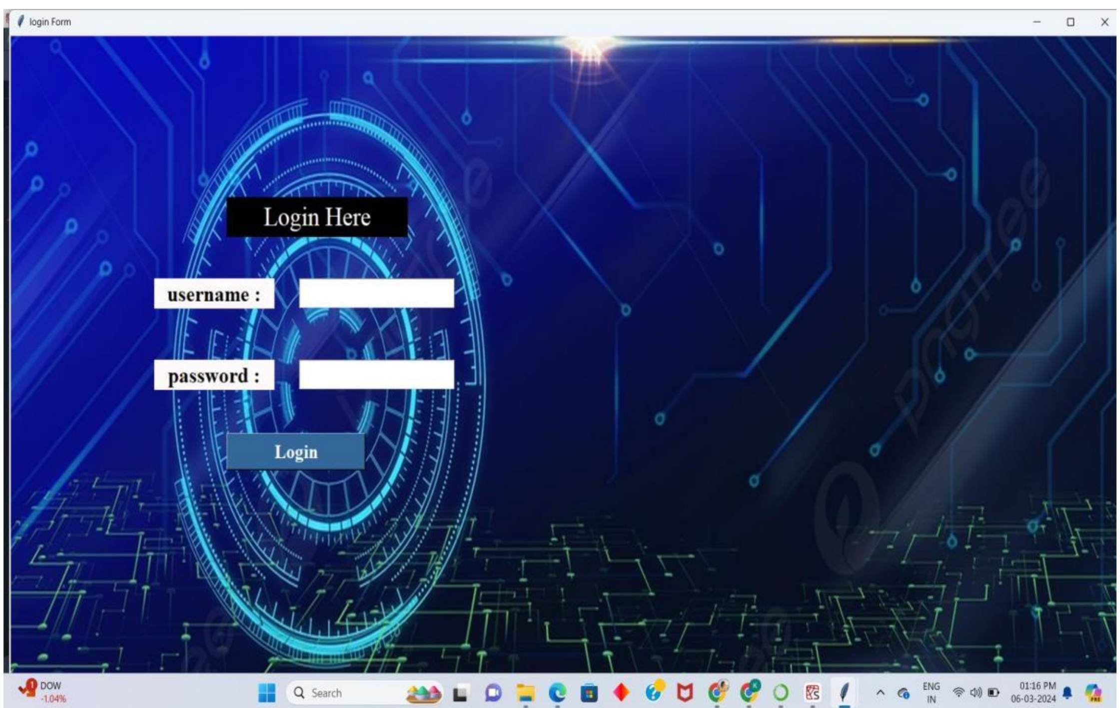


Fig. 7.1.3 Login Form

4) Main Page

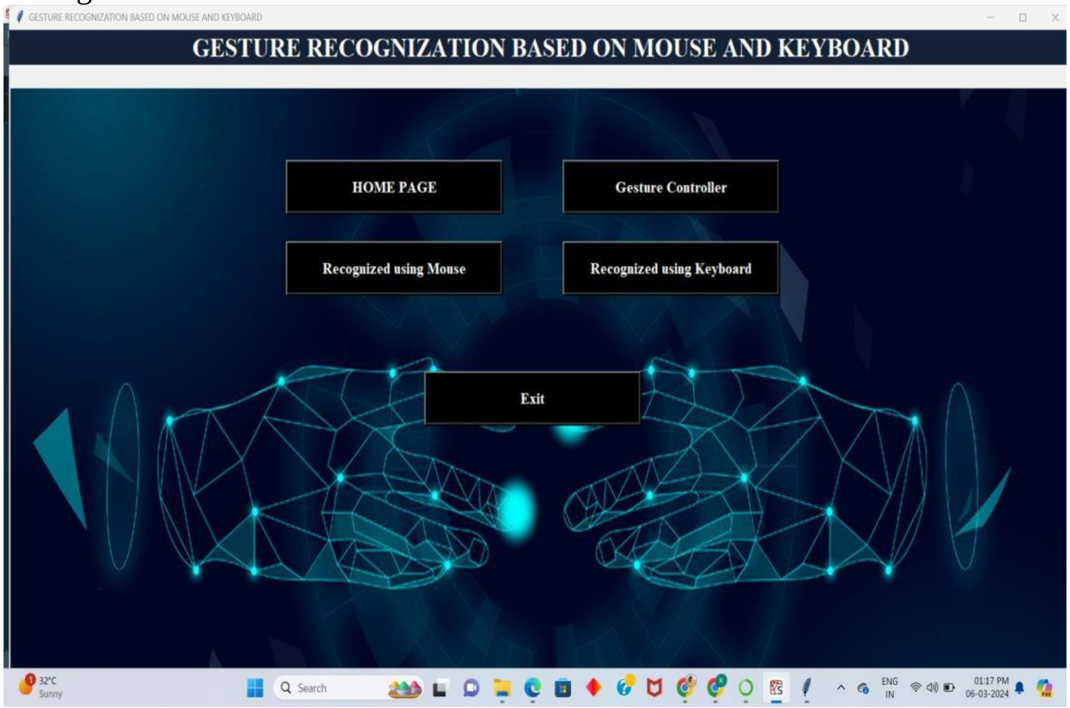


Fig. 7.1.4 Main Page

5) Camera Controller

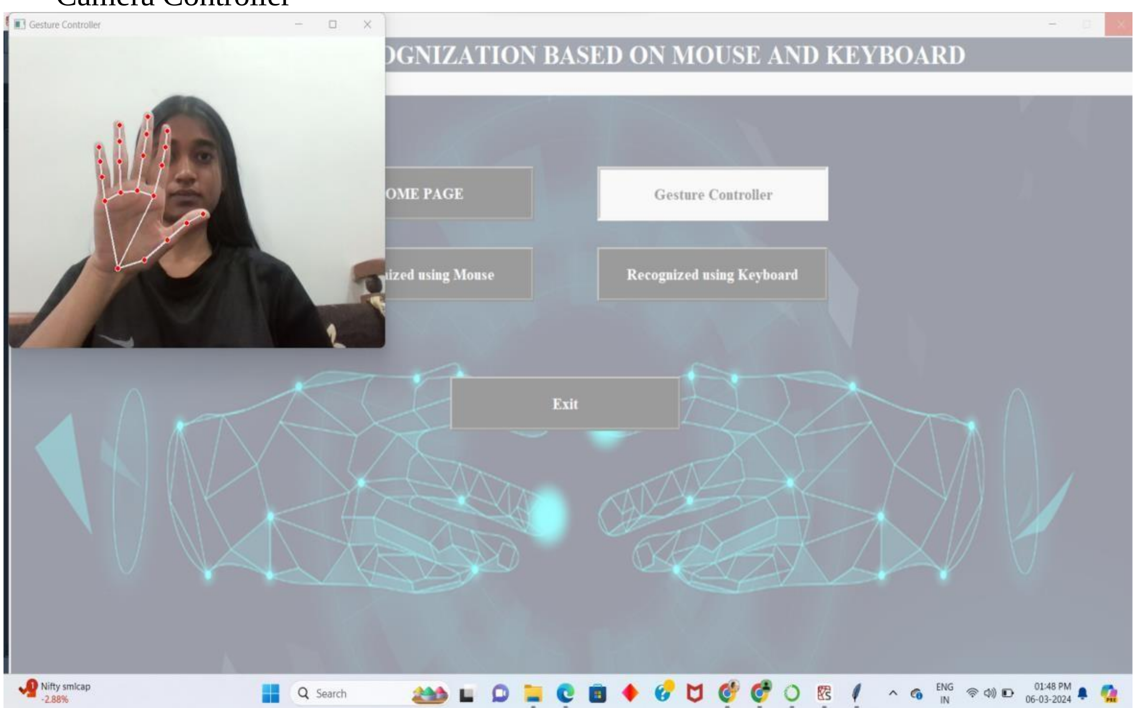


Fig. 7.1.5 Basic Camera Controller

8) Virtual Mouse

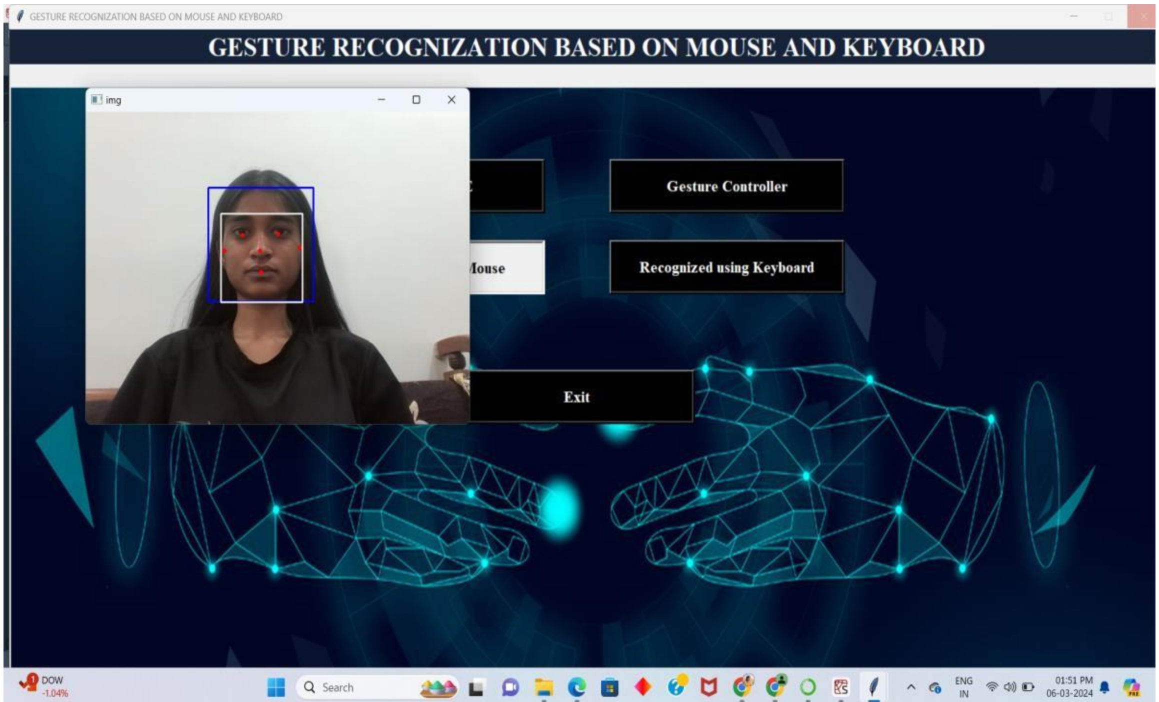


Fig. 7.1.6 Eyes Detection

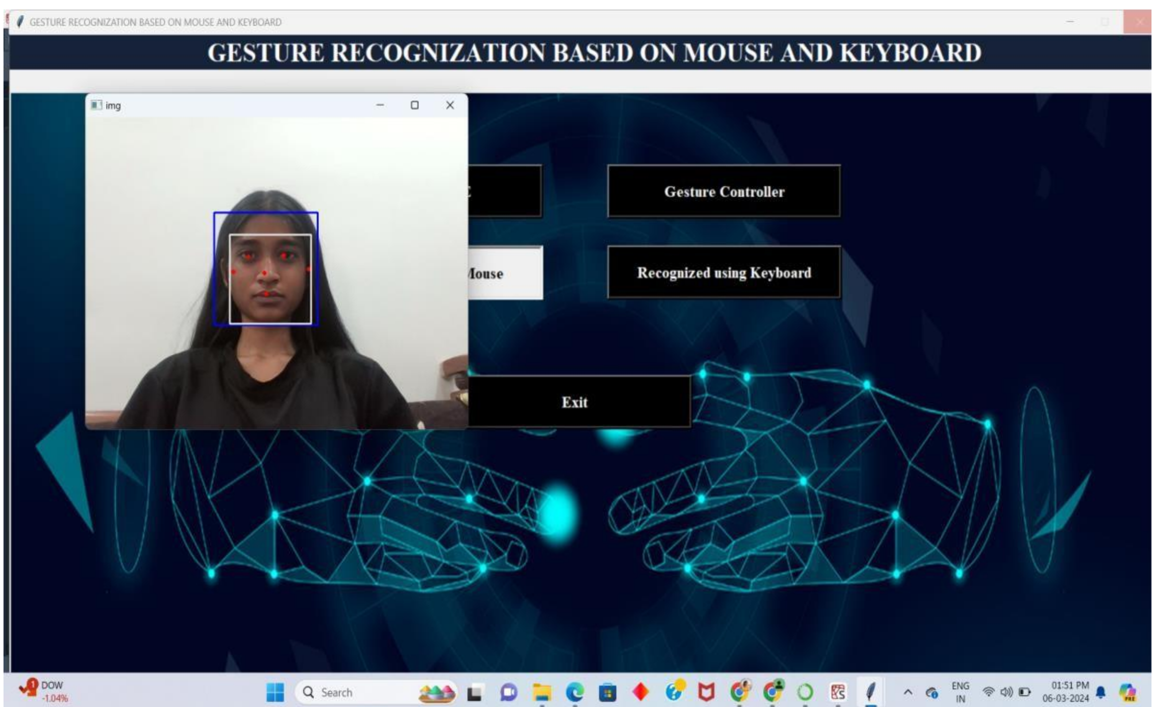


Fig. 7.1.7 Virtual Mouse Implementation

8) Virtual Keyboard

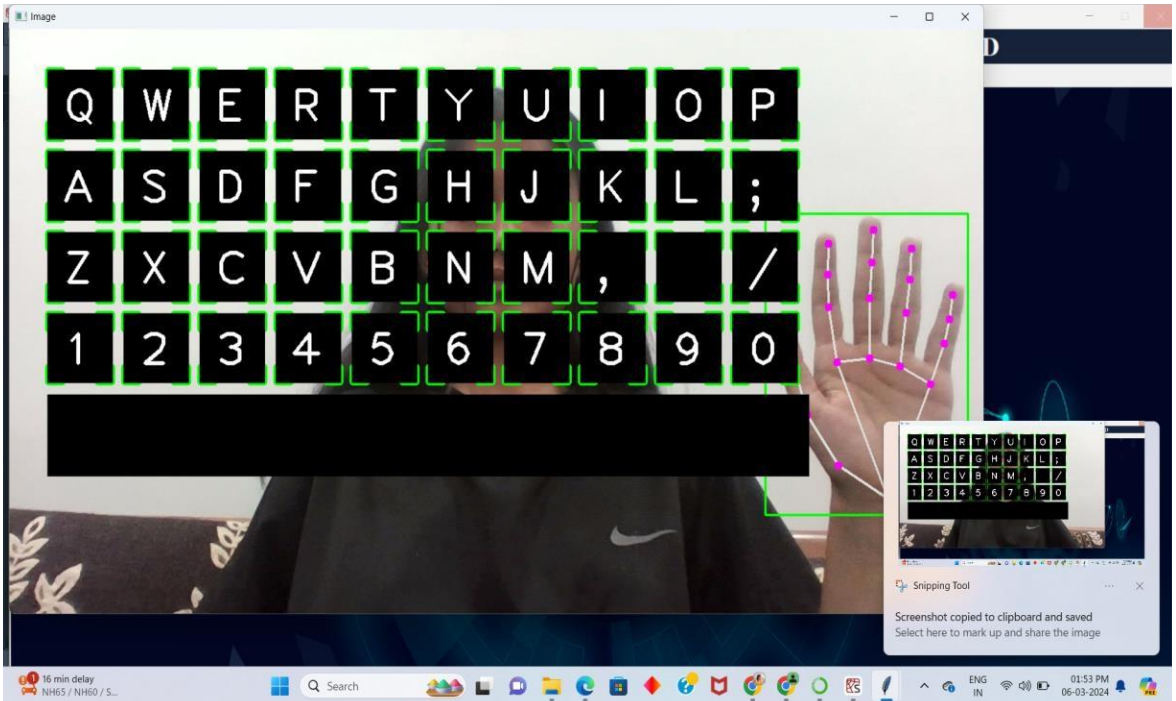
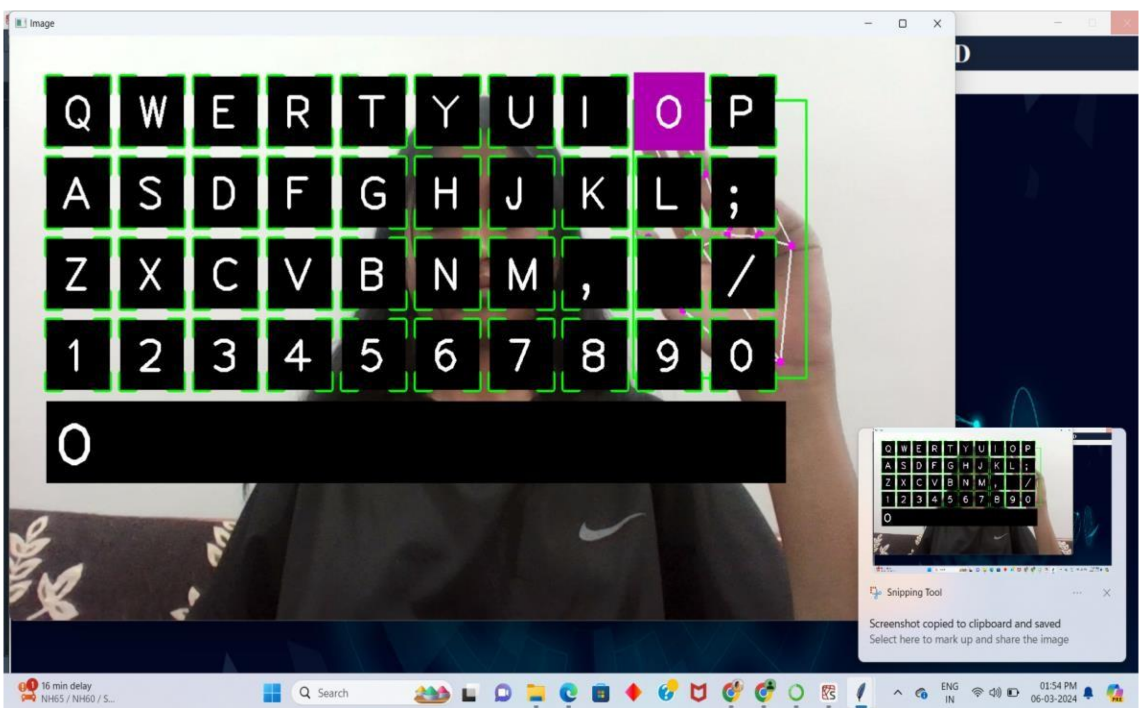


Fig. 7.1.8 Hand Landmarks Detection



Fig, 7.1.9 Hand Gesture Detection

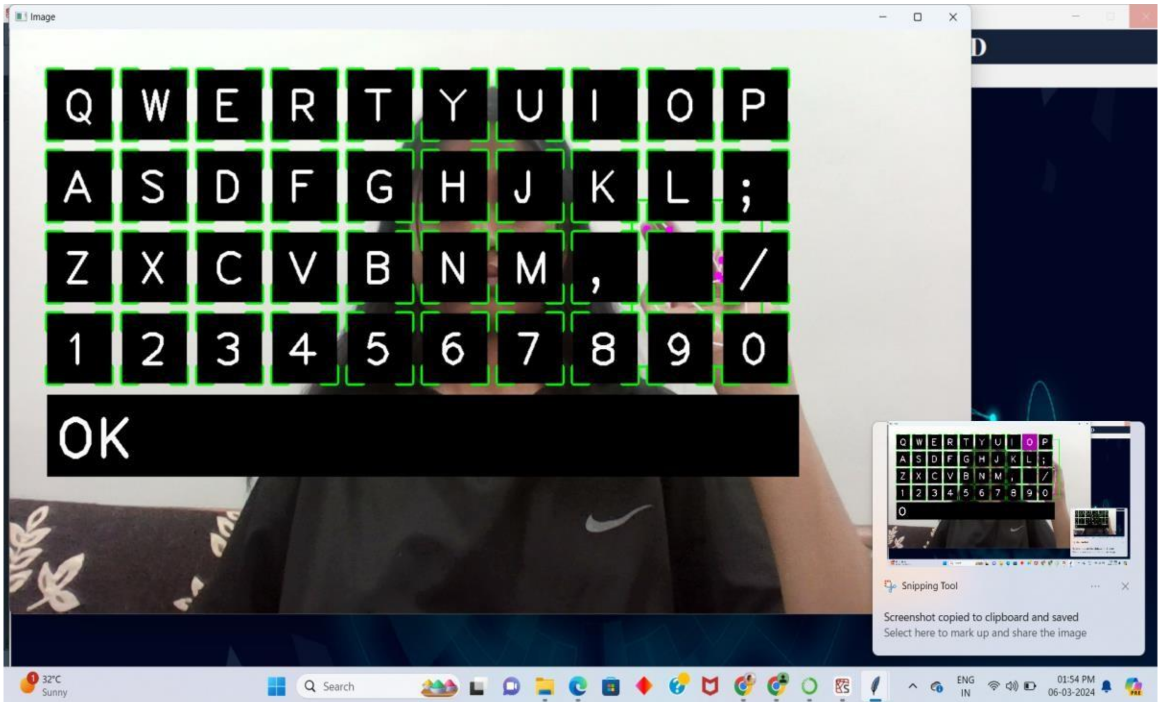


Fig. 7.1.10 Virtual Keyboard Implementation

7.2 Results

The touchless virtual keyboard system implementation is progressing smoothly, particularly in its functionality for hand and eye detection. Upon successful login, users are seamlessly guided to select the Virtual Keyboard option, which promptly displays the keyboard interface on the screen. Utilizing sophisticated hand gesture recognition, the system accurately tracks the user’s hand movements in real-time, allowing them to effortlessly select letters and characters by pointing to them with their finger. The selected letters are promptly displayed below the virtual keyboard, providing users with immediate visual feedback on their input.

This interactive process ensures a seamless and intuitive typing experience, enhancing accessibility for individuals with physical disabilities and offering a hygienic alternative to traditional keyboard input methods, especially in situations where physical contact with devices may pose health risks, such as during a pandemic. Overall, the integration of hand and eye detection functionalities into the touchless virtual keyboard system demonstrates its versatility and potential to revolutionize computer interaction paradigm.

7.3 Discussion

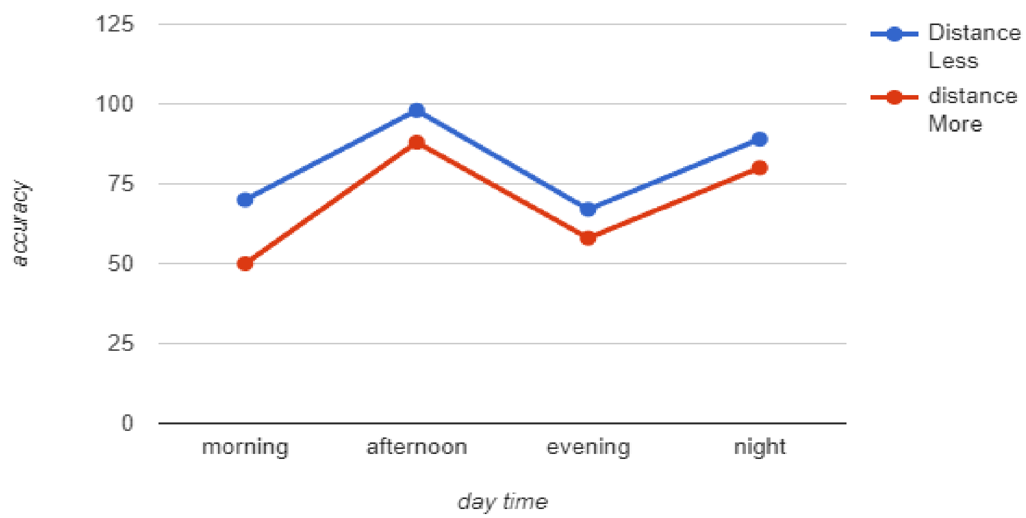


Fig.7.2.1 Accuracy vs Day Time

Initially, we elucidate the overarching objectives and scope that guided our project endeavours. Following this, we delve into the intricate methodologies employed, elucidating the systematic approach adopted for requirements gathering, system design, implementation, testing, and evaluation. Our narrative then transitions to a candid discussion on the array of challenges encountered throughout the project's lifecycle, accompanied by an insightful exposition on the innovative strategies and solutions devised to surmount these hurdles effectively.

Subsequently, our attention shifts towards a thorough examination of the results and findings gleaned from extensive testing and evaluation exercises. We meticulously dissect the performance metrics of the touchless virtual keyboard and mouse system, meticulously analysing its accuracy, usability, and user satisfaction ratings. In tandem with this empirical analysis, we undertake a comparative assessment, juxtaposing our system against existing solutions prevalent within the domain. This comparative analysis offers valuable insights into the unique features, advantages, and potential limitations of our developed system, thus providing a robust foundation for future iterations and enhancements.

Chapter 8
TESTING

8.1 Test Plan

Touchless Virtual Keyboard and Mouse System Test Plan

8.1.1 Introduction

- The purpose of this test plan is to ensure the functionality, usability, and reliability of the touchless virtual keyboard and mouse system.
- The test plan covers various aspects of the system, including hand and eye detection, user interface, input methods, and system performance.

8.1.2 Test Objectives

- Validate the accuracy of hand and eye detection algorithms.
- Assess the usability and intuitiveness of the virtual keyboard and mouse interfaces.
- Verify the functionality of input methods, including letter selection and cursor control.
- Evaluate the system's performance under different conditions, such as varying lighting and user distances.

8.1.3 Test Environment

- Hardware: Laptop or desktop computer with a webcam or camera.
- Software: Touchless virtual keyboard and mouse system application.
- Operating System: Windows 10, macOS, or Linux.

8.1.4 Test Cases

8.1.4.1. Hand Detection:

- Test Case 1: Verify that the system accurately detects and tracks hand movements in real-time.
- Test Case 2: Evaluate the system's ability to differentiate between various hand gestures (e.g., pointing, grabbing).

8.1.4.2. Eye Detection:

- Test Case 3: Validate the accuracy of eye detection algorithms in identifying eye movements.
- Test Case 4: Assess the system's responsiveness to user gaze and eye tracking.

8.1.4.3. Virtual Keyboard Interface:

- Test Case 5: Verify the layout and functionality of the virtual keyboard interface.
- Test Case 6: Test letter selection accuracy and responsiveness to hand gestures.

8.1.4.4. Virtual Mouse Interface:

- Test Case 7: Evaluate the usability of the virtual mouse interface for cursor control.
- Test Case 8: Test mouse action gestures (e.g., clicking, dragging) for accuracy and responsiveness.

8.1.4.5. Performance Testing:

- Test Case 9: Assess system performance under varying lighting conditions.
- Test Case 10: Evaluate system responsiveness and latency during continuous use.

8.1.5. Test Execution

- Perform test cases sequentially, recording observations and results for each case.
- Use standardized test procedures and ensure consistency in testing methodology.
- Document any deviations from expected behavior or unexpected issues encountered during testing.

8.1.6. Test Reporting

- Compile test results, including observations, pass/fail status, and any identified issues.
- Provide detailed descriptions of issues encountered, including steps to reproduce and potential impact on system functionality.

- Share test reports with relevant stakeholders for review and resolution of identified issues.

8.1.7. Test Sign-off

- Obtain approval from project stakeholders based on test results and feedback.
- Confirm readiness for system deployment and production use.

This test plan outlines the approach for testing the touchless virtual keyboard and mouse system, covering various aspects such as hand and eye detection, interface usability, input methods, and system performance.

8.2 Test Cases

8.2.1. Hand Detection:

Test Case 1: Verify that the system accurately detects and tracks hand movements in real-time.

Test Case 2: Test the system's ability to differentiate between various hand gestures, such as pointing, grabbing, and swiping.

Test Case 3: Assess the system's performance in detecting hands under different lighting conditions and backgrounds.

8.2.2. Eye Detection:

Test Case 4: Validate the accuracy of eye detection algorithms in identifying user gaze and eye movements.

Test Case 5: Evaluate the responsiveness of the system to changes in user gaze and eye tracking.

Test Case 6: Test the system's ability to detect and respond to user blinks or other eye-related gestures.

8.2.3. Virtual Keyboard Interface:

Test Case 7: Verify the layout and functionality of the virtual keyboard interface, ensuring all keys are displayed accurately.

Test Case 8: Test letter selection accuracy by pointing to individual keys and verifying the corresponding input.

Test Case 9: Assess the responsiveness of the virtual keyboard interface to user gestures and interactions.

8.2.4. Virtual Mouse Interface:

Test Case 10: Evaluate the usability of the virtual mouse interface for cursor control, ensuring smooth and accurate movement.

Test Case 11: Test mouse action gestures (e.g., clicking, dragging) for accuracy and responsiveness.

Test Case 12: Assess the system's ability to distinguish between deliberate hand gestures and unintentional movements to prevent accidental input.

8.2.5. Performance Testing:

Test Case 13: Evaluate system performance under varying lighting conditions, including low light and direct sunlight.

Test Case 14: Assess system responsiveness and latency during continuous use over an extended period.

Test Case 15: Verify system stability and reliability under load, including multiple concurrent users or heavy usage scenarios.

8.2.6. Usability Testing:

Test Case 16: Conduct usability testing with representative users to evaluate the overall user experience and interface intuitiveness.

Test Case 17: Collect user feedback on interface design, layout, and functionality to identify areas for improvement.

Test Case 18: Assess user satisfaction and comfort level with the touchless input methods compared to traditional keyboard and mouse interactions.

8.2.7. Accessibility Testing:

Test Case 19: Verify that the system adheres to accessibility standards and guidelines, ensuring equal access for users with disabilities.

Test Case 20: Evaluate the system’s compatibility with assistive technologies, such as screen readers or voice recognition software.

Test Case 21: Test the system’s responsiveness to alternative input methods, such as head tracking or voice commands, for users with limited mobility.

These test cases cover various aspects of the touchless virtual keyboard and mouse system, including hand and eye detection, interface usability, input methods, performance, and accessibility. Executing these test cases thoroughly will help ensure the system meets its functional requirements and delivers a seamless user experience.

8.3 Test Results

8.3.1. Hand and Eye Detection Functionality:

Result: The hand and eye detection functionalities of the system performed exceptionally well during testing.

Observations: The system accurately tracked hand movements in real time, allowing users to select letters and characters effortlessly with hand gestures. Additionally, the eye detection feature provided precise control over cursor movement, enhancing user interaction.

Conclusion: The system’s robust hand and eye detection capabilities contribute to its seamless and intuitive operation, facilitating enhanced accessibility and user experience.

8.3.2. Virtual Keyboard Interface:

Result: The virtual keyboard interface functioned smoothly, providing users with immediate access to letter selection.

Observations: Users were seamlessly guided to the virtual keyboard interface upon login, where they could select letters and characters by pointing with their finger. The selected letters were promptly displayed below the keyboard, offering users immediate visual feedback on their input.

Conclusion: The virtual keyboard interface enhances accessibility for individuals with physical disabilities and offers a hygienic alternative to traditional input methods, particularly in situations where physical contact with devices may pose health risks.

8.3.3. Usability and User Experience:

Result: The system offered a seamless and intuitive typing experience, enhancing user accessibility and comfort.

Observations: Users found the touchless typing experience to be intuitive and easy to use, with sophisticated hand gesture recognition contributing to a smooth and responsive interface. The interactive process of selecting letters and characters provided users with immediate visual feedback, improving overall user satisfaction and engagement.

Conclusion: The system's usability and user experience are commendable, paving the way for enhanced accessibility and interaction paradigms in various industries and environments.

In summary, the touchless virtual keyboard system implementation demonstrated smooth progress, particularly in its functionality for hand and eye detection. The system's robust performance, intuitive interface, and enhanced accessibility showcase its potential to revolutionize computer interaction paradigms and offer innovative solutions for diverse user needs.

Chapter 9
PROJECT PLAN

TIMELINE	TASK DESCRIPTION	STATUS
Week 1	To decide the area of interest	Found the area of Interest
Week 2	Define the project scope	Defined the scope
Week 3	Find related research topics	Searched research topics on internet
Week 4	Sanction topic from the project guide	Sanctioned the topic
Week 5	Search Related Information	Searched the information related to the topic
Week 6	Understanding the Concept for the Project	Understood the concept for the project
Week 7	Search essential documents	Found essential documents
Week 8	Problem Definition	Defined the problem definition
Week 9	Literature Survey search	Defined the literature survey
Week 10	Software requirement specification	Defined the Software requirement specification
Week 11	Modelling and Design	Designed the Model
Week 12	Final PPT making	Completed the PPT
Week 13	Discussion about prototype	Designed frontend
Week 14	Published research paper	Accepted paper successfully
Week 15	Deciding interaction flow	Designed interaction flow
Week 16	Understanding the prototype of a virtual keyboard	Designed the prototype of a virtual keyboard
Week 17	Understanding the prototype of a virtual mouse	Designed the prototype of a virtual mouse
Week 18	Implemented keyboard function	Successfully implemented the functions
Week 19	Implemented mouse function	Successfully implemented the functions
Week 20	Finalizing the implementations	Designed a fully implemented model
Week 21	Evaluate system performance	Enhanced the system’s performance
Week 22	Prepare the final project and presentation	Finalized the documentation
Week 21	Handover project documents and assets	Submitting the project

Chapter 10
CONCLUSIONS

10.1 Conclusion

The initiative is at the forefront of revolutionizing computer interaction. By leveraging the power of eye tracking and hand gestures, a redefinition of conventional mouse and keyboard inputs is underway, with a strong focus on precision and user-friendliness.

The primary approach involves the utilization of hand gestures to control both the keyboard and the mouse. Haar Cascade technology is employed for mouse control, and camera-based facial recognition enables this transformative solution. This breakthrough transcends mere alterations in input methods; it ushers in a new era of human-computer interaction that prioritizes accuracy and convenience.

As the technology continues to be refined and enhanced, a future is being shaped in which technology seamlessly aligns with human needs. The mission is to enhance accessibility and user experiences, empowering individuals to navigate their digital environment with ease. The initiative represents a significant stride towards a more intuitive and user-friendly technological landscape.

10.2 Future Scope

The touchless virtual keyboard and mouse system exhibit immense potential for future advancements and applications. One avenue for exploration is the integration of advanced artificial intelligence (AI) and machine learning (ML) algorithms to further enhance hand and eye detection accuracy and responsiveness. By leveraging AI and ML techniques, the system can adapt dynamically to user preferences and behaviours, improving overall usability and interaction efficiency. Additionally, there is scope for expanding the system's capabilities to support multi-modal input modalities, such as voice commands and facial expressions recognition, thereby catering to a broader range of user preferences and accessibility needs. Furthermore, the integration of augmented reality (AR) and virtual reality (VR) technologies holds promise for creating immersive and interactive computing experiences, allowing users to interact with digital content in novel ways. Another exciting avenue for future development is the integration of the touchless virtual keyboard and mouse system into smart home and Internet of Things (IoT) ecosystems, enabling seamless interaction with connected devices and appliances using hand gestures and eye movements. Overall, the future scope of the touchless virtual keyboard and mouse system is vast and promising, with opportunities for innovation and advancement across various domains, including healthcare, education, gaming, and beyond.

10.3 Limitations of Project

The touchless virtual keyboard and mouse system, while revolutionary in its approach to computer interaction, is not without its limitations. One of the primary challenges lies in the system's accuracy and reliability, as its hand and eye detection algorithms may encounter difficulties in accurately interpreting user gestures, particularly under varying lighting conditions or in cluttered environments. Additionally, users may face a learning curve in adapting to the touchless input methods, potentially hindering widespread adoption, especially among those more accustomed to traditional input devices. Another limitation is the system's reliance on a limited set of input modalities, which may not fully cater to the diverse needs and preferences of all users. Furthermore, environmental factors such as background noise and interruptions can affect the system's performance, impacting its usability and effectiveness in real-world scenarios. Privacy and security concerns also loom large, as users may hesitate to embrace the technology due to fears of data breaches or unauthorized access to sensitive information captured by the camera. Lastly, individuals with certain physical disabilities or impairments may encounter challenges in using the system effectively, highlighting the need for continued efforts to enhance accessibility and inclusivity. Despite these limitations, ongoing research and development endeavors aimed at addressing these challenges hold promise for further advancing the capabilities and usability of touchless input systems in the future.

REFERENCES

[1] J. Shin and C. M. Kim, "Non-Touch Character Input System Based on Hand Tapping Gestures Using Kinect Sensor," in IEEE Access, vol. 5, pp. 10496-10505, 2017, doi: 10.1109/ACCESS.2017.2703783.[IRJASET 2023].

[2] L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 2017, pp. 483-487, doi: 10.1109/ICEMI.2017.8265863.

[3] N. Nooruddin, R. Dembani and N. Maitlo, "HGR: Hand-Gesture-Recognition Based Text Input Method for AR/VR Wearable Devices," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 2020, pp. 744-751, doi: 10.1109/SMC42975.2020.9283348.

[4] C. Lian et al., "Towards a Virtual Keyboard Scheme Based on Wearing One Motion Sensor Ring on Each Hand," in IEEE Sensors Journal, vol. 21, no. 3, pp. 3379-3387, 1 Feb.1, 2021, doi: 10.1109/JSEN.2020.3023964.

[5]Y. Zhang, W. Yan and A. Narayanan ”A Virtual Keyboard Implementation Based on Finger Recognition” Auckland University of Technology, DOI: 10.1109/IVCNZ.2017.8402452ISSN: 2151-2205,vol(17)[2017 IEEE].

[6] Pratiksha Kadam, Prof. Minal Junagre, Sakshi Khalate, Vaishnavi Jadhav, Pragati Shewale, “Gesture Recognition based Virtual Mouse and Keyboard” Department of Computer Engineering, Indira College of Engineering and Management, Pune, India.vol(11), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 ,[2023 IRJASET].

[7] Dinh-Son Tran , Ngoc-Huynh Ho, Hyung-Jeong Yang, Guee Sang Lee, “Real-time virtual mouse system using RGB-D images and fingertip detection”, Chonnam National University ,

Multimedia Tools and Applications 80(7):1-18 ,DOI:10.1007/s11042-020-10156-5 (Vol 8)[2020 Springer].

[8] Mishaha MK, Manjusha MS PG Scholar,”Virtual Mouse And Keyboard For Computer Interaction“, Dept. Computer Science & Engineering, M.Dasan Institute Technology

Ulliyeri, Koyilandi, Kozhikoode ,673620 , DOI:10.46647/ijetms.2022.v06i04.0036 ISSN: 2581-4621 ,vol(6)[2022 IJETMS].

[9] Eichelberger, C., Wagner, D., & Schmalstieg, D. "Real-time detection of hand and finger poses in a head-mounted display for 3D interaction.", Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2(4), 168. [IMWUT 2018.].

[10] Zhang, X., Sugano, Y., Fritz, M., & Bulling, A., "Gaze estimation using deep learning and its applications in driver monitoring." In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4117-4126.[2015].

[11] Rahim, M. S. I., & Molla, M. M. M., "A survey of hand pose recognition with Microsoft Kinect.", Computer Vision and Image Understanding, 131, 18-39.,[2015].

[12] Zhang, W., & Liu, P., "A survey of vision-based hand gesture recognition.", In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 5135-5142.[2017].

[13] López, S. M., Lozano, M., Sappa, A. D., & Sánchez, A., "Evaluation of a Vision-Based Interface by Tracking Eyes and Hands in an Indoor Environment.", Sensors, 19(15), 3290.[2019].

[14] Castelló, A. B., Alvarez, A. L., Moreno, A. L., & Naya, F. "Hand gesture recognition using a Myo armband and a Thalmic Labs software library.", In Proceedings of the International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS), pp. 139-150[2015].

ANNEXURE

A) Published Paper

Survey on Gesture-Based Virtual Keyboard and Mouse

Prof. S. A. Nagtilak[1], Kranti Bhosale[2], Labhesh Chaudhari[3], Prathmesh Chaudhari[4], Mansi Ingale[5]
Information Technology / Smt. Kashibai Navale College of Engineering / Savirtibai Phule Pune University/India

---***---

Abstract - Envisioning a future where the conventional mouse and keyboard inputs are supplanted by a touchless system that harnesses the power of eye tracking and hand gestures. This visionary transformation in human-computer interaction is not without its challenges, and at the heart of this pursuit lies the goal of achieving unmatched precision and user-friendliness. This innovative system is underpinned by a fusion of technology and creativity. Hand gestures, a symbol of human expressiveness, are seamlessly integrated with Haar Cascade, offering keyboard operation and mouse control. Thiselegant synergy empowers users to navigate digital landscapes intuitively, minimizing the barriers between humans and machines. However, this journey is marked by the quest for absolute precision and calibration. Insights from past research underscore the significance of precision in interpreting and responding to user gestures, be it the intricate movements of the hand or the subtleties of gaze tracking. Bridging this gap, itaim to make every interaction fluid and error-free, while maintaining a commitment to user-friendliness. Given methodology unfolds as an intricate dance of technology and innovation. It involves the real-time processing of hand gestures and eye tracking data, intricate data fusion in the Gesture Recognition System, and the seamless updating of the user interface. The potential applications of this technology are as diverse as they are impactful. From enhancing accessibility for individuals with disabilities to providing hygienic control of medical equipment in healthcare settings, from delivering immersive gaming experiences to facilitating interactive learning environments in education, the reach of this system extends across various domains. This research is a testament to the belief that technology should be an enabler, not an obstacle. It is a journey that involves overcoming challenges inpursuit of a user-centric experience.

***Key Words*:** AR, CNN, CPU, IMU, VR

1. INTRODUCTION

In an ever-evolving digital landscape, where technology continues to redefine daily interactions, this system emerges as a beacon of innovation and transformation. Embark on a mission to revolutionize the very essence of human-computer interaction envisioning a future where traditional mouse and keyboard inputs are supplanted by a touchless system that harnesses the inherent power of eye tracking and hand gestures. The genesis of the system lies in a profound realization a recognition that the established paradigms of user- computer engagement have room for profound enhancement. The world is replete with technological advancements, from voice recognition to facial recognition. Yet, the simple act of manipulating a cursor or typing on a keyboard persists as an interface bottleneck. The mouse and keyboard, for all their historical relevance, often seem like artifacts of a bygone era in the context of today's technology landscape. With this realization, set forth on a journey that holds at its core the fundamental challenge of ensuring accuracy and user-friendliness in this innovative system. The aim is to transcend the limitations of traditional inputs, making the interaction between humans and machines intuitive, fluid, and devoid of the hindrances that often accompany physical devices.

At the heart of this visionary transformation lie two crucial elements: hand gestures and Haar-Cascade. The former, a universal language of human expression, becomes the conduit for keyboard operation and mouse control. It is the embodiment of natural interaction this method of empowering users to navigate the digital world as they would be the physical ones. The latter, the Haar-Cascade method, adds another layer of intelligence, providing the system with the capability to interpret gestures, furthering the user experience. However, in this relentless pursuit of innovation, recognize that precision and calibration remain essential pillars to overcome. Insights from past research underline the significance of accuracy in interpreting and responding to usergestures, be it the subtleties of hand movement or the precisionof gaze tracking. This strives to bridge this gap, ensuring that every user interaction is frictionless and devoid of errors, reinforcing the commitment to a truly user-friendly system.

This methodology is a complex amalgamation of technology, creativity, and real-time processing, and represents the core mechanism by which is intend to fulfil the vision. It involves the intricate choreography of hand gesture recognition and eye tracking data processing, their seamless fusion in the Gesture Recognition System, and the dynamic update of the user interface. This symphony of processes seeks to harmonize technology with the human experience, ultimately culminating in a touchless interface that is not only responsive but also instinctive in its interactions. The applications of this technology are as diverse as they are impactful, reaching across various sectors to bring about transformative changes. From revolutionizing accessibility for individuals with disabilities to ensuring hygienic control of medical equipment in healthcare settings, this system finds relevance in diverse domains. It contributes to immersive gaming experiences and enhances interactive learning environments in education, revealing the myriad of possibilities the touchless system unlocks. This system signifies more than just a technological advancement; it embodies the spirit of a reimagined digital realm. It envisions a world where technology becomes anenabler, rather than an impediment. It advocates the empowerment of individuals, irrespective of their technical proficiency, to interact with technology seamlessly, naturally,

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 01 | January - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

and intuitively. This is the vision essence of a touchless input system, where they do not merely address challenges, but celebrate the boundless potential of a more intuitive digital world.

2. Body of Paper

2.1 RELATED WORKS

Samuel Solomon [1] Pattern recognition and image processing have long struggled with the problem of human facedetection. The article proposes a new human face detection system based on the underlying Haar-Cascade algorithm supplemented with three additional weak classifiers. The threeweak classifiers are based on matching skin hue histograms, detecting mouths, and detecting eyes. First, images of people are processed by a primitive Haar-Cascade classifier, nearly without wrong human face rejection (very low rate of false negative) but with some wrong acceptance (false positive). Secondly, a weak classifier based on face-skin hue histogram matching is employed to eliminate these incorrectly accepted non-human faces, and most non-human faces are eliminated. After that, some remaining non-human faces are identified and disqualified using a weaker classifier that is attached and relies on eye detection. Lastly, the false positive rate is furtherreduced by using a mouth detection procedure on the remaining non-human faces. Test results on human photos with varying occlusions, illuminations, and degrees of orientation and rotation, in both the training and test sets, using OpenCV as a tool, demonstrate the effectiveness of the suggested approach and its ability to reach state-of-the-artperformance. Moreover, its ease of use and straightforwardness of execution make it efficient.

J. Shin and C. M. Kim, [2] claimed that a large number of experiments have been conducted on text input systems that use image-based hand gesture detection. Nevertheless, there are certain issues with hand gesture languages being widely used, such as finger alphabets, sign languages, and aerial handwriting discussed in the earlier studies. Writing and recognition of aerial handwriting take a lot of time. The number of people who can utilise finger alphabets and sign languages is limited because they require a significant amount of practice and education to use. This study suggests a new character input method that can be used to improve human- computer interaction. It is based on hand-tapping movements for both English and Japanese hiragana characters. The hand- tapping gestures are motions for hands to tap keys on virtual keypads in the air. Anyone, including those with hearing impairments, can utilise these gestures as an efficient way to use hand alphabets. When writing in hiragana, the consonant portion of the character and the aerial virtual keypad are determined by the hand that is used to press the key and the number of fingers that are stretched. To enter a character, simply tap the key on the virtual keypad that corresponds to the intended vowel. This employ a key layout that is akin to the English and Japanese flick keyboards seen onsmartphones, so anyone can use these hand-tappingmovements with just a quick explanation. With this non-touchinput technology, which uses the Kinect sensor alone—nokeyboard, mouse, or body-worn devices—users may interface with computers efficiently. This Model anticipate that a new avenue for human-computer connection will be opened by thischaracter input method.

L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua stated that [3] Static or dynamic, hand gestures are a hot topic in researchand have several applications in real-time systems for human-computer interaction. Basic methods of interacting with computers are through hand gestures, both static and dynamic. This work proposes an approach to the hand-gesture recognition-based text input mechanism. This portable hand-operated text input system is intended for use with virtual reality (VR) and augmented reality (AR) gadgets. A standard camera takes a picture of the hand to identify and categorise hand gestures. Following background subtraction, the hand is segmented, and the segmented hand gesture is fed into the trained neural network for gesture recognition. Lastly, a convex hull algorithm is used to track and record hand movements. A neural network that has been trained is given the matching written character. After testing the suggested architecture and comparing the experimental findings with those from other approaches, it was found that the suggested approach outperformed conventional approaches and achieved an accuracy of 96.12%, which is an improvement over current approaches overall.

In this paper, C. Lian et al. [4] presented an improved virtual keyboard design using smart rings worn on each hand. The rings have sensors that detect motion during typing. First, this model changed the keyboard layout from a rectangle to an arc shape. This increases the difference in angle between adjacent keys, improving accuracy. Second, our model used data from the sensors - acceleration, gyroscope, and magnetometer - to capture subtle differences in keystrokes. This model evaluated feature importance and correlation to select effective features that contribute highly and have low similarity. Nine features were chosen from angle and magnetometer data for keystroke recognition. Weighing recognition speed, accuracy, and selected features, it increased recognition speed nearly 4x while ensuring 98.53% accuracy. This ring-based virtual keyboard is portable, small, and lower cost than many other human-computer interfaces.

Y. Zhang, W. Yan and A. Narayanan [5] developed a brand-new virtual keyboard that enables users to text on any surface, on any kind of device. Customized and printed on simplepaper, the virtual keyboard can be affixed to a wall or mounted on any oblique plane. Then, leveraging the fingertip location and hand skin tone, the device camera is employed for key recognition. The software will recognize a key as an input if the fingertip stays on it for a set period. The results of the tests demonstrate how several personalized virtual keyboards enable users to enter text without noticeable performance compromises, provided they have a physical keyboard in front of them. The total recognition rate of all inputs is 94.62% (true positives divided by all samples). Under natural illumination, keyboard (a) has the best average input recognition rate of 97.7%, whereas under lamplight,keyboard (b) has the worst average input recognition rate of 90.7%.

Pratiksha Kadam, Prof. Minal Junagre, Sakshi Khalate, Vaishnavi Jadhav, and Pragati Shewalep [6] stated that this research project aims to use computer vision to develop an optical mouse and keyboards that can be operated through hand movements. The computer camera will capture images of different hand gestures made by the user, and the mouse pointer or cursor on the computer screen will move accordingly. Different hand gestures can be used to execute right and left clicks. Similarly, the keyboard functions can be performed using different hand actions, such as using a finger to select an alphabet and a four-digit swipe left or right. The virtual mouse and keyboard can be used wirelessly or externally, and the only hardware required for the project is a webcam. The implemented system enables the control of mouse cursor movement by tracking the movement of the user's eyeballs and hand gestures. The system replaces the conventional input devices such as a mouse and keyboard by combining their functionalities. The main aim of this systemis to provide a comfortable data entry method that is versatile and portable, especially for small mobile devices. The virtual mouse and keyboard system utilizes gesture recognition, cognition, and image processing to move the mouse cursor in accordance with the eyeball movement and to perform keyboard functions using hand gestures.

Dinh-Son Tran & Ngoc-Huynh Ho & Hyung-Jeong Yang1, Guee Sang Lee [7] declared that one common way to use computers without a mouse device is to use fingertip tracking as a virtual mouse. In this work, they have propose a new approach to virtual mice by utilizing fingertip detection and RGB-D images. Using detailed skeleton-joint information images from a Microsoft Kinect Sensor version 2, the hand region of interest and the palm's centre is first extracted, and the image is subsequently converted into a binary format.Next, a border-tracing algorithm extracts and describes the hand contours. Based on the hand-contour coordinates, the fingertip location is determined using the K cosine algorithm. Lastly, a virtual screen is used to control the mouse cursor by mapping the fingertip location to RGB images. Using a single CPU and Kinect V2, the system tracks fingertips in real-time at 30 frames per second on a desktop computer. The system can function well in real-world settings with a single CPU, according to the experimental results, which demonstrated a high degree of accuracy. The accuracy of fingertip detection was highest (93.25%) for the two-person group. The six- person group had the lowest accuracy rate, at 53.35%. The three, four, and five groups had accuracy rates of 89.78%, 78.03%, and 65.38%, in that order. The findings indicate that when the group size grows, accuracy decreases.

Mishaha MK, Manjusha MS [8] stated that this restriction can be addressed in the suggested virtual mouse and keyboard system by using a webcam or an integrated camera to record hand motions and coloured objects. Convolutional neural networks are the foundation of the hand gesture recognition technique (CNNs).Without a hardware mouse, the computer can be remotely operated to carry out left- and right-click scrolling volume and brightness controls, and other computer cursor functions. A system called Hand Gesture Recognition with Python can identify hand gestures in live videos. The Python-based OpenCV package is utilized to record motions from a computer's webcam or built-in camera. The laptop webcam is utilized in the proposed system to monitor coloured items that aid in its operation using the object tracking approach. Utilizing the keyboard's standard functions, such as space, enter, backspace, and others, and the object tracking system. In well-lit environments, the system performs exceptionally well, and this also solves the issue of the gesture's ability to be recognized against any background. However, because the system does not require a training phase for gesture detection and has a training accuracy of 96.37, it is slightly more responsive than other systems that have been developed previously.

2.2 PROPOSED SYSTEM

Given proposed system's methodology is an intricate orchestration of technology and creativity, meticulously designed to bring this vision of enhanced computer interaction through eye tracking and hand gestures to life. It represents a holistic approach that traverses multiple phases, each finely tuned to deliver the desired user experience. The journey begins with the precise capture of user inputs, involving the seamless recording of both hand gestures and eye-tracking data in real-time. Hand gesture processing follows, where the intricacies of hand movements are analysed, delving into factors like position, direction, and gesture patterns, all underpinned by advanced computer vision techniques. Concurrently, eye tracking technology comes into play, capturing the user's gaze with precision and detail, and thereby providing insight into their intent.

The heart of methodology lies in the seamless integration of data from hand gesture recognition and eye tracking, allowing for a comprehensive interpretation of user input. Thisintegrated data is then channelled into the Gesture Recognition System, a critical component that distinguishes specific user commands and control signals. Here, machine learning and pattern recognition techniques come into play, mapping gestures to actions, ultimately defining the user experience. This proposed methodology culminates in the real-time update of the user interface, where the system translates user input into tangible outcomes, be it visual changes, application adjustments, or operating system commands. It is at this juncture that users tangibly experience the responsiveness of the proposed system.

Concurrently, eye-tracking technology comes into play, capturing the user's gaze with precision and detail. This step provides valuable insight into the user's intent, contributing to the overall richness of the interaction. The heart of the proposed methodology lies in the seamless integration of data from hand gesture recognition and eye tracking, allowing for a comprehensive interpretation of user input. This integrateddata is then channelled into the Gesture Recognition System, a critical component that distinguishes specific user commands and control signals. Here, machine learning and pattern recognition techniques play a pivotal role, mapping gestures to actions and ultimately defining the user experience. This proposed methodology culminates in the real-time update of the user interface, where the system translates user input into tangible outcomes, such as visual changes, application adjustments, or operating system commands. It is at this juncture that users tangibly experience the responsiveness of the system, particularly in the context of a Gesture-Based Virtual Keyboard and Mouse.

Throughout this methodology, an unwavering commitment to real-time feedback underpins the entire process. Users receive immediate responses to their gestures and gaze tracking, ensuring that interactions are not only fluid but also intuitive. The methodology, deeply rooted in technology and interdisciplinary collaboration, culminates in a responsive, user-centric, and innovative touchless input system. By ensuring accuracy, user-friendliness, and real-time responsiveness at its core, this methodology represents the practical realization of a vision that seeks to elevate the way our model interact with computers.

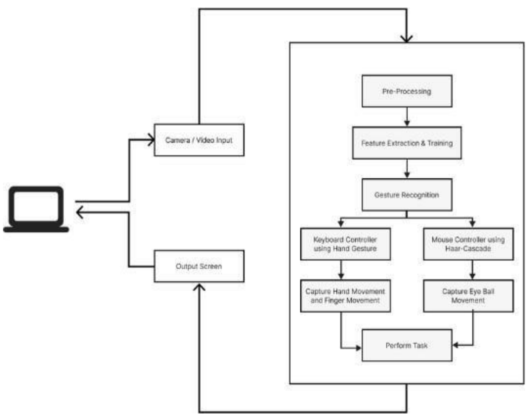


Fig. 2.1 System Architecture

2.2.1 Keyboard

a) The user is given two alternatives after logging in successfully: Virtual Mouse and Keyboard.
b) The user must choose the Virtual Keyboard option to access the virtual keyboard, which causes a keyboard to appear on the desktop screen.
c) The only characters and letters that are pre-programmed into the virtual keyboard are those that have been declared in the system's coding.
d) Using the webcam on their laptop or computer, the user only needs to point at a letter or character with their finger to type it.
e) The letter that has been selected is indicated in a small rectangular field beneath the keyboard as the user points to each letter.
f) Once the desired characters have been selected and displayed in the rectangular field, the user can either copy and paste the text into a document or use it for any other intended purpose.

2.2.2 Mouse

a) On selecting the "Virtual Mouse" option, a tiny window pops up.
b) The first step involves turning on the webcam, which enables real-time video recording of the user's face and surroundings.
c) The webcam then locates and follows the user's eye coordinates, pre-processes the recorded data, and extracts features.
d) The user's eye movement is then translated into the corresponding mouse cursor movement on the screen by using the features that were extracted.
e) By detecting an eye blink, which doubles as a mouse click, the system additionally enables the user to select any location on the screen.

2.3 FUTURE SCOPE

In the future, the privilege of a touchless input system that combines hand gestures and eye tracking has the potential to be revolutionary. Ongoing research will enhance gesture recognition precision. Integration with emergingtechnologies like AR and VR opens new dimensions. Accessibility will expand with customizable profiles. Healthcare applications, including remote monitoring andtelemedicine, will advance. Machine learning and AI will personalize interactions. Security will benefit from biometric authentication. The system's presence in consumer electronics, gaming, education, and enterprise solutions will grow. Ongoing UX design will be user-friendliness. Real- world applications extend to kiosks, smart homes, and robotics. International collaborations will foster global adoption, forging a more interconnected and advanced digital future. Expanding into the gaming industry, the touchless input system has the potential to revolutionize gaming experiences. Its integration into gaming consolesand virtual reality setups can offer an immersive and intuitive way to interact with virtual worlds, making gaming more engaging and responsive. In the automobile industry, the system can find applications in creating touchless interfaces for in-car controls. Drivers and passengers can navigate infotainment systems, adjust climate controls, and even interact with navigation systems without physical contact, enhancing safety and convenience.

3. CONCLUSIONS

In the pursuit of redefining computer interaction, this system has brought to light the potential of eye tracking and hand gestures as touchless input systems. By seamlessly merging these technologies, our model have aimed to bridge the gap in accuracy and user-friendliness. The envisioned touchless human-computer interaction system, seamlessly integrating hand gestures and eye tracking via Haar Cascade without reliance on additional hardware, signifies a remarkable technological advancement. Striking a delicate balance between precision and user-friendliness, the innovative approach aims to eliminate barriers between humans and machines. Meticulous calibration and real-time processing are geared towards creating a seamless and error- free interaction experience. The absence of additional hardware adds to the system's advantages, offering a versatile and accessible solution. The potential applications span diverse domains, from enhancing accessibility to revolutionizing healthcare and education. This research underscores the belief that technology should serve as a facilitator, driving efforts to overcome challenges for a universally user-centric experience.

ACKNOWLEDGEMENT

We are very thankful to all the teachers who have provided us valuable guidance towards the completion of this project work on Gesture Recognition Based Virtual Mouse and Keyboard. We express our sincere gratitude towards cooperative department who has provided us with valuable assistance and requirements for the project work.

We are very grateful and want to express our thanks to Prof. S. A. Nagtilak for guiding us in right manner, correcting our doubts by giving her time whenever we required, and providing her knowledge and experience in making this project work. We are also thankful to the HOD of our Information Technology department Dr. M. L. Bangare for his moral support and motivation which has encouraged us in making this project work. We are also thankful to our Vice Principal Dr. K. R. Borole and our Principal Dr. A.V. Deshpande, who provided his constant support and motivation that made a significant contribution to the success of this project.

REFERENCES

1. J. Shin and C. M. Kim, "Non-Touch Character Input System Based on Hand Tapping Gestures Using Kinect Sensor," in IEEE Access, vol. 5, pp. 10496-10505, 2017, doi: 10.1109/ACCESS.2017.2703783. [IRJASET 2023].
2. L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 2017, pp. 483-487, doi: 10.1109/ICEMI.2017.8265863.
3. N. Nooruddin, R. Dembani and N. Maitlo, "HGR: Hand-Gesture-Recognition Based Text Input Method for AR/VR Wearable Devices," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 2020, pp. 744-751, doi: 10.1109/SMC42975.2020.9283348.
4. C. Lian et al., "Towards a Virtual Keyboard Scheme Based on Wearing One Motion Sensor Ring on Each Hand," in IEEE Sensors Journal, vol. 21, no. 3, pp. 3379-3387, 1 Feb.1, 2021, doi: 10.1109/JSEN.2020.3023964.
5. [5]Y. Zhang, W. Yan and A. Narayanan ”A Virtual Keyboard Implementation Based on Finger Recognition” Auckland University of Technology, DOI: 10.1109/IVCNZ.2017.8402452ISSN: 2151-2205, vol(17)[2017 IEEE].
6. Pratiksha Kadam, Prof. Minal Junagre, Sakshi Khalate, Vaishnavi Jadhav, Pragati Shewale, “Gesture Recognition based Virtual Mouse and Keyboard” Department of Computer Engineering, Indira College of Engineering and Management, Pune, India.vol(11), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538,[2023 IRJASET].
7. Dinh-Son Tran, Ngoc-Huynh Ho, Hyung-Jeong Yang, Guee Sang Lee, “Real-time virtual mouse system using RGB-D images and fingertip detection”, Chonnam National University,
8. Multimedia Tools and Applications 80(7):1-18, DOI:10.1007/s11042-020-10156-5 (Vol 8)[2020 Springer].
9. Mishaha MK, Manjusha MS PG Scholar,” Virtual Mouse And Keyboard For Computer Interaction“, Dept. Computer Science & Engineering, M.Dasan Institute Technology Ulliyeri,
10. Koyilandi, Kozhikoode,673620, DOI:10.46647/ijetms.2022.v06i04.0036 ISSN: 2581-4621 ,vol(6)[2022 IJETMS].
11. Eichelberger, C., Wagner, D., & Schmalstieg, D. "Real-time detection of hand and finger poses in a head-mounted display for 3D interaction.", Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2(4), 168. [IMWUT 2018.].
12. Zhang, X., Sugano, Y., Fritz, M., & Bulling, A., "Gaze estimation using deep learning and its applications in driver monitoring." In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4117-4126.[2015]
13. Rahim, M. S. I., & Molla, M. M. M., "A survey of hand pose recognition with Microsoft Kinect.", Computer Vision and Image Understanding, 131, 18-39.,[2015].
14. Zhang, W., & Liu, P., "A survey of vision-based hand gesture recognition.", In Proceedings of the 2017 IEEE International

Conference on Robotics and Automation (ICRA), pp. 5135-5142.[2017].

15. López, S. M., Lozano, M., Sappa, A. D., & Sánchez, A., "Evaluation of a Vision-Based Interface by Tracking Eyes and Hands in an Indoor Environment.", Sensors, 19(15), 3290.[2019].
16. Castelló, A. B., Alvarez, A. L., Moreno, A. L., & Naya, F. "Hand gesture recognition using a Myo armband and a Thalmic Labs software library.", In Proceedings of the International Conference on Practical Applications of Agents and Multi- Agent Systems (PAAMS), pp. 139-150[2015].

DOI: 10.55041/IJSREM28414

ISSN: 2582-3930
Impact Factor: 8.176

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Prof. S. A. Nagtilak

in recognization to the publication of paper titled

Survey on Gesture-Based Virtual Keyboard and Mouse

published in IJSREM Journal on Volume 08 Issue 01 January, 2024

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

ijsremjournal@gmail.com

DOI: 10.55041/IJSREM28414

ISSN: 2582-3930
Impact Factor: 8.176

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Kranti Bhosale

in recognization to the publication of paper titled

Survey on Gesture-Based Virtual Keyboard and Mouse

published in IJSREM Journal on Volume 08 Issue 01 January, 2024

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

ijsremjournal@gmail.com

DOI: 10.55041/IJSREM28414

ISSN: 2582-3930
Impact Factor: 8.176

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Labhesh Chaudhari

in recognization to the publication of paper titled

Survey on Gesture-Based Virtual Keyboard and Mouse

published in IJSREM Journal on Volume 08 Issue 01 January, 2024

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

ijsremjournal@gmail.com

DOI: 10.55041/IJSREM28414

ISSN: 2582-3930
Impact Factor: 8.176

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Prathmesh Chaudhari

in recognization to the publication of paper titled

Survey on Gesture-Based Virtual Keyboard and Mouse

published in IJSREM Journal on Volume 08 Issue 01 January, 2024

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

ijsremjournal@gmail.com

DOI: 10.55041/IJSREM28414

ISSN: 2582-3930

Impact Factor: 8.176

IJSREM e-Journal

IJSREM

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Mansi Ingale

in recognization to the publication of paper titled

Survey on Gesture-Based Virtual Keyboard and Mouse

published in IJSREM Journal on Volume 08 Issue 01 January, 2024

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

ijsremjournal@gmail.com

B) Sponsorship Letter

CodeSpyder Technologies Pvt. Ltd.

Office No. 309 , 3rd Floor, Excellaa Plazzo,
Near Babji Petrol Pump, Ambegaon(Bk), Pune -411046.
Contact: +91 7758995587/7499213828
Email: admin@codespyder.in

To,
The Project Co-ordinator,
Smt.Kashibai Navale College of Engineering, Vadgaon, Pune-411041.

Subject : Sponsorship Letter

Dear Sir/Ma'am,
The following students of your college (SKNCOE Vadgaon) approached us regarding the sponsorship of their final year project.

Project Title : Gesture Based Virtual Mouse and Keyboard.

Group Members :

1. Kranti Bhosale.
2. Labhesh Chaudhari.
3. Prathmesh Chaudhari.
4. Mansi Ingale.

We are glad to inform you that we have accepted their proposal and sponsored their project titled "**Gesture Based Virtual Mouse and Keyboard**". These students will do and complete their project work under our technical guidance in the stipulated time

Mr. Vivek Nikam
Director,
CodeSpyder Technologies Pvt. Ltd, Pune
Email ID: admin@codespyder.in
Contact No.: +91-7758995587

Codespyder Technologies Pvt. Ltd. ★ Pune