**HAND GESTURE-BASED VIRTUAL MOUSE**

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

**BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE & ENGINEERING**

**BY**

**HRITIK JOSHI(EN18CS301101)**

**NITIN WAYBHASE(EN19CS3L1014)**

Under the Guidance of

**MR. DHARMENDRA MANGAL**



**Department of Computer Science & Engineering**

**Faculty of Engineering**

**MEDI-CAPS UNIVERSITY, INDORE- 453331**

**JAN-MAY 2022**

**HAND GESTURE-BASED VIRTUAL MOUSE**

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

**BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE & ENGINEERING**

**BY**

**HRITIK JOSHI(EN18CS301101)**

**NITIN WAYBHASE(EN19CS3L1014)**

Under the Guidance of

**MR. DHARMENDRA MANGAL**



**Department of Computer Science & Engineering**

**Faculty of Engineering**

**MEDI-CAPS UNIVERSITY, INDORE- 453331**

**JAN-MAY 2022**

**Report Approval**

The project work **“HAND GESTURE-BASED VIRTUAL MOUSE "is** hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in; but approve the “Project Report” only for the purpose for which it has been submitted.

**Internal Examiner**

Name: Mr.Ratnesh Litoriya

Designation: Assistant Professor

Affiliation: Medi-Caps University, Indore

Name: Mr.Dharmendra Mangal

Designation: Assistant Professor

Affiliation: Medi-Caps University, Indore

**External Examiner**

Name:

Designation:

Affiliation:

**Declaration**

I/We hereby declare that the project entitled **“HAND GESTURE-BASED VIRTUAL MOUSE”** submitted in partial fulfillment for the award of the degree of Bachelor of Technology/Master of Computer Applications in ‘COMPUTER SCIENCE AND ENGINEERING DEPARTMENT completed under the supervision of **MR. Dharmendra Mangal, Assistant Professor Computer Science & Engineering,** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, I/we declare that the content of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

**HRITIK JOSHI(EN18CS301101)**

**DATE - 29/04/22**

**NITIN WAYBHASE (EN19CS3L1014)**

**DATE – 29/04/22**

**Certificate**

I, **Prof**. **Dharmendra Mangal** certify that the project entitled **“HAND GESTURE-BASED VIRTUAL MOUSE ”**submitted in partial fulfillment for the award of the degree of Bachelor of Technology/Master of Computer Applications by **HRITIK JOSHI AND NITIN WAYBHASE** is there cord carried out by him/them under my/our guidance and that the work has not formed the basis of award of any other degree elsewhere.

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Mr. Ratnesh litoriya |  | Dr. Pramod S. Nair |
| Computer Science & Engineering |  | Head of the Department |
| Medi-Caps University, Indore |  | Computer Science & Engineering |
|  |  | Medi-Caps University, Indore |

**Acknowledgements**

I would like to express my deepest gratitude to Honorable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) D. K. Patnaik,** Vice Chancellor, Medi-Caps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) Suresh Jain,** Dean, Faculty of Engineering, Medi-Caps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr.Pramod S. Nair** for his continuous encouragement for betterment of the project.

I express my heartfelt gratitude to my Guide, **MR. Dharmendra mangal,** Assistant Professor, Department of Computer Science & Engineering, MU, without whose continuous help and support, this project would ever have reached to the completion.

I would also like to thank to my team students name who extended their kind support and help towards the completion of this project.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

**HRITIK JOSHI (EN18CS301101)**

**NITIN WAYBHASE (EN19CS3L1014)**

B.Tech. IV Year

Department of Computer Science & Engineering

Faculty of Engineering

Medi-Caps University, Indore

**ABSTRACT**

In today’s world, we tend to see a lot of development happening in the field of Technology. Today’s technology is combined with the technique known as computer science. This paper is additionally supported little a part of AI. This paper presents a Hand gesture-based virtual mouse on our computer’s window exploitation camera & handling the complete system by simply moving your fingers. Using finger detection ways for fast camera access and easy computer program makes it additional simply accessible. The system is employed to implement a motion tracking mouse. this technique reduces the employment of any physical mouse which saves time and additionally reduces effort.

Keywords: Image processing; Hand gesture; mediapipe; Virtual mouse control; Finger detection.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **Page No.** |
|  |  |  |
|  | Report Approval | ii |
|  | Declaration | iii |
|  | Certificate | iv |
|  | Acknowledgement | v |
|  | Abstract | vi |
|  | Table of Contents | vii |
|  | List of figures | viii |
|  | List of tables | ix |
|  | Abbreviations | x |
|  | Notations & Symbols | xi |
| Chapter 1 | Introduction | 10-13 |
|  | 1.1 Introduction | 10 |
|  | 1.2 Literature Review | 11 |
|  | 1.4 Problem statement | 12 |
|  | 1.4 Objectives | 13 |
|  | 1.5 Research Design | 13 |
|  | 1.6 Chapter Scheme | 13 |
| Chapter 2 | Experimental Setup | 13-16 |
|  | 2.2 Procedure Adopted | 16 |
|  | 2.3 System feasibility | 16 |
| Chapter 3 | System Requirement Specification | 17-20 |
|  | 2.1 Existing System | 17 |
|  | 2.2 Proposed System | 17 |
|  | 2.3 Hardware Specification | 18 |
|  | 2.4 Software Specification | 18 |
|  | 2.5 Functional Requirement | 19 |
|  | 2.6 Non-Functional Requirement | 20 |
| Chapter 3 | System Analysis & Design | 21-28 |
|  | 3.1 Use case Diagram | 21 |
|  | 3.2 Statechart Diagram | 22 |
|  | 3.3 Class Diagram | 25 |
|  | 3.4Data flow Diagram | 25 |
|  | 3.5 Activity Diagram | 26 |
|  | 3.6 Sequence Diagram | 27 |
|  | 3.7 Architectural Design | 28 |
| Chapter 4 | Implementation | 28-29 |
|  | 4.1 Procedural Description | 28 |
|  | 4.2 Module Description | 29 |
| Chapter 5 | Testing | 31-32 |
|  | 5.1 Unit Testing | 31 |
|  | 5.2 Integration Testing | 31 |
|  | 5.3 Validation Testing | 31 |
|  | 5.4 White Box Testing | 31 |
|  | 5.5 Performance Testing | 32 |
| Chapter 6 | Algorithm and technique | 32-36 |
| Chapter 7 | Results and Discussions | 36 |
| Chapter 8 | Summary and Conclusions | 37 |
| Chapter 9 | Future scope | 37 |
|  | Appendix | 39 |
|  | Bibliography |  |
|  | List of Publications (If any) |  |
|  | Reprints of publications (If any) |  |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **Page No.** |
| Figure 1 | Block diagram of hand gesture-based virtual mouse. | 13 |
| Figure 2 | Co-ordinates or land marks in the hand | 13 |
| Figure 3 | For moving the cursor we will use two fingers. | 32 |
| Figure 4 | To drag the file we will use all the fingers | 33 |
| Figure 5 | For volume up and down (scroll up and down )we will use one finger and one thumb. | 33 |
| Figure 6 | For brightness up and down we will use one finger and the thumb | 34 |
| Figure 7 | For the left click, we will use one finger. | 34 |
| Figure 8 | 7 For the Right click, we will use one finger. | 35 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **Page No.** |
| 1 | Result | 36 |
| 2 | Test Case | 37 |

# 

# LIST OF ABBREVIATIONS

**HGR** – Hand Gesture Recognition

**RGB** – Red, Green, and Blue

**HSB** – Hue, Saturation, and Brightness

**OpenCV** – Open Source Computer Vision

**MIPF** – Marvin Image Processing Framework

**FR** – Functional Requirements

**NFR** – Non-Functional Requirements

**ROI** – Region of Interest

**CPU** – Central Processing Unit

**GUI** – Graphical User Interface

**API** – Application Programming Interface

**JRE** – Java Runtime Environment

**CHAPTER 1**

**INTRODUCTION**

The hand gesture-based virtual mouse is a software program that allows users to provide mouse inputs to a device without using a physical mouse. This research presents a computer creative hand gesture-based virtual mouse device that produces, using hand gestures and hand tip detection, for performing mouse activities on the computer. The major purpose of the suggested device is to perform laptop mouse cursor functions using a webcam or a built-in digital camera within the laptop rather than a conventional mouse device. A computer web camera is used in conjunction with various image processing techniques to create a hand gesture-based digital mouse.

 In this research hand movements of a user are used as mouse inputs. A web camera is a set of cameras that indefinitely take photos, and most laptops now include them. Webcams have also been used by security applications that use face recognition to harness the potential of face detection. To fully utilize the power of a system camera, it can be used for Vision-based CC is frequently used, which eliminates the need for such a computer mouse and mouse pad. They can also be used in HCI applications such as motion controllers and sign language databases, which can benefit greatly from using a system camera. A wireless mouse is used to control a system camera. A wireless mouse or a Bluetooth mouse takes several components to work, including a mouse, a dongle, and a battery, but in this project, the client will operate the computer mouse using hand gestures using a built-in camera or a web camera.

The hand gesture-based digital mouse gadget was created using the open Python programming language, as well as OpenCV, a computer vision package that is employed inside the system. As a result, the Media Pipe package is used to track the hands and monitor the end of the thumbs in this edition. The gadget camera gathers and approaches the collected frames in this system, identifying various hand motions and hand tip gestures, and then performing real mouse functionalities.

**1.2. Literature Survey**

As trendy technology of human pc interactions become necessary in our everyday lives, sorts of a mouse of quite shapes and sizes were unread, from an inform workplace mouse to a hard-core diversion mouse. However, there are some limitations to this hardware as they're not as environmentally friendly as it appears. as an example, the physical mouse needs a flat surface to work, to not mention that it needs an explicit space to today to utilize the functions offered. moreover, a number of this hardware are buttery useless once it involves activities with the computer remote because of the cable lengths imitations, rendering it inaccessible.

Sande et al.[1] suggested The present virtual mouse control system comprises generally mouse operations that control the hand gesture-based virtual mouse, left-click, right-click, and scrap-down, among other things, using a hand gesture detection system. Although there are several hand recognition systems, the one they chose was static hand recognition, which is merely a recognition of the shape created by the hand and therefore the definition of action for each shape made, which is confined to a few defined actions and generates a lot of confusion. There are more and additional alternatives to using a mouse as technology progresses.

Agrawal et al.[2] suggested The main goal of this paper is to control any computer vision algorithm-based application running on a computer using two of the most important modes of interaction: head and hand. The video input stream hand is segmented, and the corresponding gesture is recognized based on the shape and pattern of a hand movement. The hidden Markov models are used for the common pre-processing of hand and head gesture virtual mouse. First, take a picture with the camera. A via-jones method is used to detect the second hand and face.

Badi. [3] suggested The basic aim of static hand gesture recognition is to identify given hand gesture data represented by specific attributes into a finite number of gesture classes. The major goal of this work is to explore the usage of two feature extraction approaches, specifically hand contouring and complex moments, to solve the problem of hand gesture detection by identifying the key benefits and drawbacks of each method.

Thakur et al.[4] suggested A hand gesture-based system to handle various mouse actions such as eft and right-clicking, scrolling up and down, and other mouse actions using hand gestures to provide interaction, additional efficiency, and reliability. This paper delineates a hand gesture-based interface for regulating a computer mouse via 2D hand gestures. Coor detection algorithms based on cameras are used to detect hand movements. This technique primarily focuses on the effective usage of a Web Camera to create a virtual device. Each input image's centroid is located. Because hand movement directly moves the centroid, it is the sensing principle for changing the pointer on a computer screen. The left and right-click scroll up down functions of a mouse are implemented by folding the first and middle fingers of the hand respectively, and developing So, comparing the length of fingers images with those in the image gives an idea about the functionality performed by the hand gesture-based virtual mouse.

Pradhan et al.[5] suggested general cursor or trackpad screen, a control system, and the act of a hand gesture control mechanism from the current system. it is not possible to use a hand gesture to access the monitor screen from a distance. The breadth is generally limited in the virtual mouse field, even though it is primarily trying to perform. The code is written in Python and uses the open-source OpenCV image processing module as was the Python-specific PyAutoGUi library to implement mouse actions. From the webcam's real-time video, just the three colored finger caps are extracted.

**1.3 Problem statement**

To operate a software method to resolve the challenge, keep in mind the issue. The goal here is to devise the most efficient process for humans to can interact with a laptop without having any physical interface with it. Many concepts had been suggested, but they all required hardware motion. Given that hand gesture and hand Tip detection areas sized to are used to handle the laptop mouse functions using a webcam or digital camera, AI digital mouse is frequently used to overcome these challenges. "Hand gesture digital mouse with digital cam" is based on the notion of using Kinect sensors with an HD camera, although the digital camera and Kinect sensor. Using a simple web digital cam, this research aims to reduce costs and improve the robustness of the suggested machine.

**1.4 Objective Of Project**

The goal of this paper is to create a computer program that uses alternative cursor control mechanisms. This paper intends to create an alternative to the typical virtual mouse system for laptops and computers. The mouse control function may be done by using a web camera and color detection to execute a hand motion virtual mouse hand tip finger. Perform the ai hand gestures of left-clicking, right-clicking, double-clicking, scrolling up and down, and dragging with a virtual mouse.

* The goal of this research is to develop a computer vision-based system for detecting, capturing, and understanding gestures.
* The task is to develop a replacement "low-cost, fast-speed, and coloration image" acquisition device.

# 1.5 Research Design

In this section, the research in each of the different algorithm designs of the hand gesture recognition (HGR) system

1.6 Chapter Scheme

The Chapter Scheme in the project report is planned to be as under:

Chapter 1 Introduction

Chapter 2 Experimental set up

Chapter 3 System Requirement Specification

Chapter 4 System Analysis and Design

Chapter 5 Implementation

Chapter 6 Testing

Chapter 7 Results and Discussions

Chapter 8 Summary and Conclusion

Chapter 9 Future Scope

**CHAPTER 2**

**EXPERIMENTAL SET-UP**

The various functions and conditions used in the system are explained in the flowchart(figure 1) of the hand gesture-based virtual mouse.

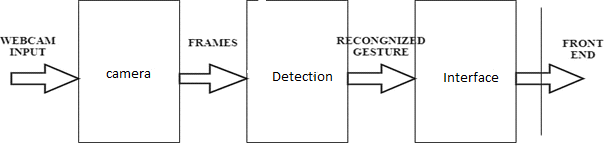


Figure 1:Block diagram of hand gesture-based virtual mouse.

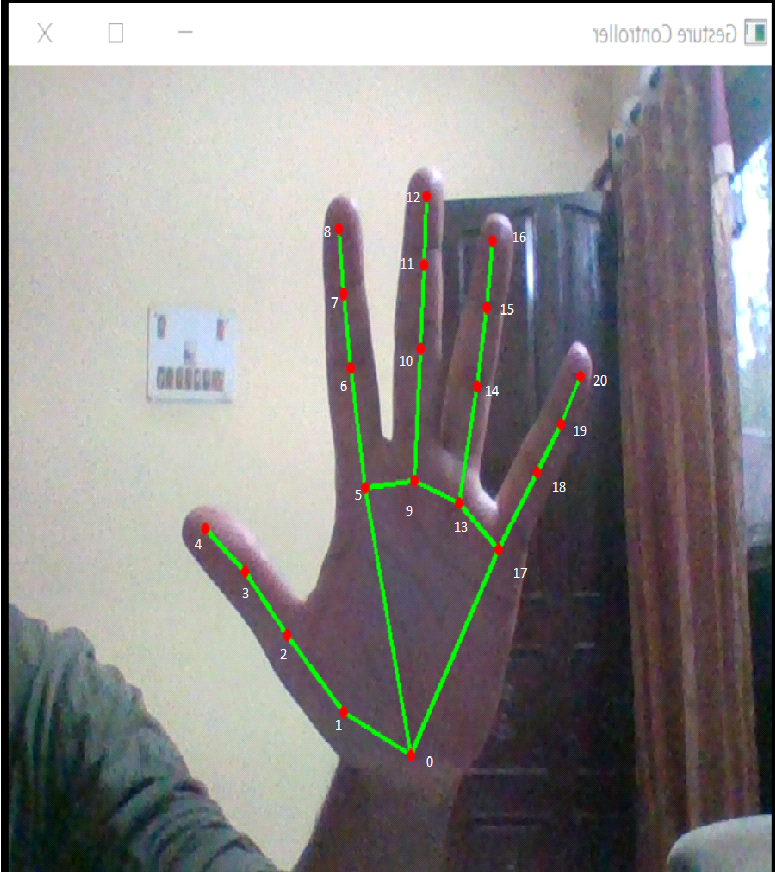


Figure 2: Co-ordinates or land marks in the hand

WRIST

THUMB\_CMC

THUMB\_MCP

THUMB\_IP

THUMB\_TIP

INDEX\_FINGER\_MCP

INDEX FINGER\_PIP

INDEX FINGER\_DIP

INDEX FINGER\_TIP

MIDDLE\_FINGER\_MCP

MIDDLE FINGER\_PIP

MIDDLE FINGER\_DIP

MIDDLE FINGER\_TIP

RING FINGER\_MCP

RING FINGER\_PIP

RING FINGER\_DIP

RING FINGER\_TIP

PINKY MCP

PINKY\_PIP

PINKY DIP

PINKY TIP

**2.1.1 The webcam Used in the Hand Gesture Based Virtual Mouse System.**

The proposed hand gesture-based virtual mouse system is based on the frames that have been captured by the camera on a laptop or PC. We are using the Python computer vision library OpenCV, the video capture object is created and the webcam will start capturing video. The system camera captures and passes the frames to the system .which is shown in figure 2.

**2.1.2 Capturing the Video and Processing**.

The hand gesture-based virtual mouse system uses the system cam where each frame will capture till the termination of the program. The video frames are processed from BGR to RGB color space to find the hands in the video frame by frame shown in figure 3.

def find Hands(self, img , draw = True):

imgRGB=cv2.cvtColor(imgcv2.COLOR\_BGR2RGB)

self. Results = self.hands.process (imgRGB)

**2.1.3 Rectangular Region for moving through the window**

The transformational formula is used by the AI virtual mouse system, to transfer the coordinates of the IP from the digital camera screen to the full-screen pc display for dominating the mouse. Once the area unit of the hand is identified, and once we realize that a finger is up for performing arts with the mouse, an oblong box is drawn about the computer window within the digital camera region, wherever we tend to move around at the window using the mouse indicator.

**2.1.4 Detecting the finger is up and performing**

The Mouse performs. We tend to detective work that finger is up misusing the tip Id of the several fingers that we tend to found misusing the MediaPipe and thus the several co-ordinates of the fingers that are up, and then the actual mouse perform is performed following that and shown in figure 4.

**2.1.5 Mouse function betting on the Hand Gesture and Hand tip Detection mistreatment pc vision for the mouse indicator on the road the pc window**

If the finger is up with tip Id = one (1) or each the finger with tip Id = one(1) and therefore the finger with tip Id = two (2) area unit up, the mouse indication is created to a makeover from around the window of the computer using the AutoPy module of Python, shown in figure 3.

**2.1.6 “For the mouse to perform Left Button Click”**

If each the index with tip Id = one and therefore the thumb-finger with tip Id = zero square measure up and therefore the distance between the 2 fingers is lesser than 30px, the computer is created to conduct a left button click using the victimization the input, which is shown in figure 7.

**2.1.7 “For the mouse to perform Right Button Click”**

If each the middle with tip Id = two(2) and therefore the thumb-finger with tip Id = zero square measure up and therefore the distance between the 2 fingers is lesser than 40px, the computer is created to conduct a right button click using the victimization the input, which is shown in figure 8.

**2.1.8 “For the Mouse to Perform Scroll up and Down”**

If both the index finger with tip Id = 1 and the thumb finger with tip Id= 0 is up and the distance between the two fingers is lesser than 30px, the computer is made to perform the scrolling up and down which is shown in figure 5.

**2.1.9 “For the Mouse to Control Brightness”**

If both the index finger with tip Id = 1 and the thumb finger with tip Id= 0 is up and the distance between the two fingers is lesser than 30px, the computer is made to perform the brightness less or more which is shown in figure 5.

**2.2 Procedures Adopted**

(a) Initialize a web camera and capture video.

(b) The system performs Hand detection.

(c) Any Hand within the image is filtered and therefore the frame is blurred.

(d) The frame is then passed to filter backgrounds and extract options.

(e) With only the image of hands stay within the image, the system can find the hand and perform pure mathematics translations.

(f) Perform an action

Mouse control mode: find hand movements and translate the coordinates to the user’s screen

Cursor movement

Left-click

Right-click

Scroll up

Scroll down

2.3 System Feasibility

This proposed system conferred a brand new approach to hand gesture-based virtual mouse management employing a combination of geometry formula and a deep learning technique to achieve detection and gesture recognition tasks. This approach exhibited not only extremely correct gesture estimates but additionally suitableness for sensible applications. The proposed technique has several benefits, for example, operating well in ever-changing light-weight levels or with advanced backgrounds, correct detection of hand gestures at an extended distance, experimental results indicating that this approach may be a promising technique for hand-gesture-based interfaces in real-time. The system isn't possible for recognizing the hand gestures of multiple people. In the long run work with hand gesture recognition, we have a tendency to shall expand our system to handle additional hand gestures and apply our technique in additional different sensible applications.

CHAPTER -3

**SYSTEM REQUIREMENT SPECIFICATION**

**3.1 Existing system**

The existing system consists of a mouse that may be either wireless or wired to manage the pointer, know we are able to use hand gestures to monitor the system. the present virtual mouse system consists of the easy mouse operation using the colored for detection that square measure captured by web-cam, thus colored fingers act as Associate in a Nursing object that the web-cam sense color like red, green, blue color to watch the system, whereas may perform basic mouse operation like minimize, drag, scroll up, scroll down, left-click right-click victimization hand gestures with none colored finger as a result of the coloring recognition system is additional versatile than the present system. within the existing system use static hand recognition like tip identification, hand shape, variety of fingers to outline action expressly, which makes a system additional advanced to grasp and tough to use.

**3.2 Proposed System**

The first stage is that object detection. The target of this stage is to find hand objects within the digital pictures or videos. several setting and image issues are required to unravel at this stage to make sure that the hand contours or regions will be extracted exactly to reinforce the popularity accuracy. Common image issues contain unstable brightness, noise, poor resolution, and distinction. the higher setting and camera devices will effectively improve these issues. However, it's laborious to regulate once the gesture recognition system is functioning within the real setting or is become a product. Hence, the image process technique could be a higher answer to unravel these image issues to construct an Associate in Nursing reconciling and sturdy gesture recognition system. The second stage is beholding. The detected hand objects are recognized to spot the gestures. At this stage, differentiated options and effective classifiers choice are serious issues in most research. The third stage is to analySe serial gestures to spot users’ instructs or behaviors.

**3.3 System feasibility**

This proposed system conferred a brand new approach to hand gesture-based virtual mouse for mouse management employing a combination of geometry algorithmic rule and a deep learning technique to achieve detection and gesture recognition tasks. This approach exhibited not only extremely correct gesture estimates but additionally quality for sensible applications. The projected technique has several blessings, for example, operating well in ever-changing light levels or with advanced backgrounds, correct detection of hand gestures at an extended distance, experimental results indicating that this approach may be a promising technique for hand-gesture-based interfaces in real-time. The system isn't possible for recognizing the hand gestures of multiple folks. In the long run, working with a hand gesture-based virtual mouse, we have a tendency to will expand our system to handle additional hand gestures and apply our technique in additional different sensible applications

**3.4 Hardware Specification:**

* Processor Intel Core
* Memory 8GB
* Graphics NVIDIA GeForce GT 720M 2GB Graphics or above
* Display 14" HD Display (15:6) Wide Screen
* Web Camera
* Communications Wi-Fi 802.11b/g/n, Ethernet Port
* Windows 10
* Webcam 1MP Fixed Focus CMOS camera on the laptop

3.5 Software Specification:

Microsoft Window 8 or above Support Architecture:

• 32bit(x86)

• 64bit(x86)

• 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit)

• 16 GB available hard disk space (32-bit)

• Spyder

• Python

• Open cv

• NumPy 1.19

3.5 Functional Requirem ent

* Face Detection

This software shall utilize a face detection system to filtrate faces from the video capturing device. By applying face detection, the system will disregard the region wherever the face is found and so reducing the number of calculation required to perform hand detection. The face detection unit is implemented with the assistance of OpenCV.

* Skin Detection Module:

This software system shall perform coloring detection and filtrate all objects that don't contain the color of skin. By filtering objects of non-skin color, the system will then use its remaining resources and target hand detection and gesture recognition

* Mouse Movement Gesture Control Mode:

After getting the location of the hand, the computer code shall use the detected location because of the mouse cursor point. Becausethe user moves his/her hands, the mouse ought to follow promptly on the screen.

.3.6 Non Functional Requirements

* Extensibility:

The software package shall be protrusile to support future developments and add-ons to the HGR software package. The gesture management module of HGR shall be extensible to permit new gesture recognition options to be additional to the system.

* Portability:

The HGR software shall be 100% portable to all or any operating platforms. Therefore, this software shouldn't depend upon the various operating system

* Performance

This software system shall minimize the number of calculations required to perform image process and hand gesture detection

* Reliability:

The HGR software system shall be operable altogether under lighting conditions. despite the brightness in the user’s operative surroundings, the program shall invariably discover the user’s hands.

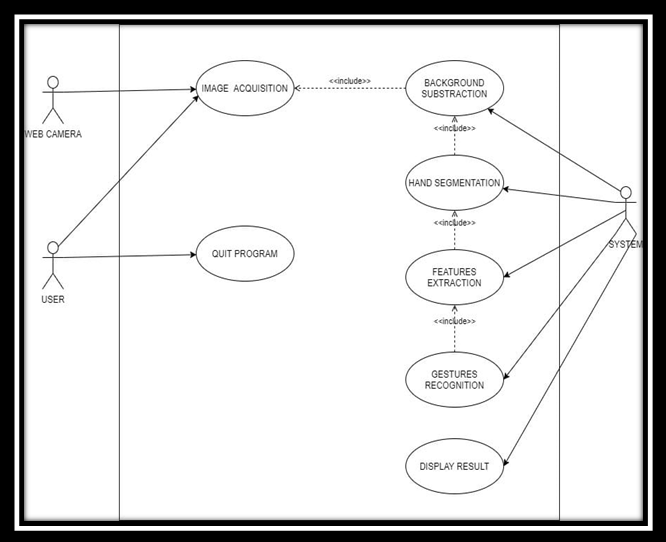
* Usability:

This software package shall be straightforward to use for all users with minimal directions. The graphical interface (GUI) shall be intuitive and intelligible by non-technical users too.

**CHAPTER 4**

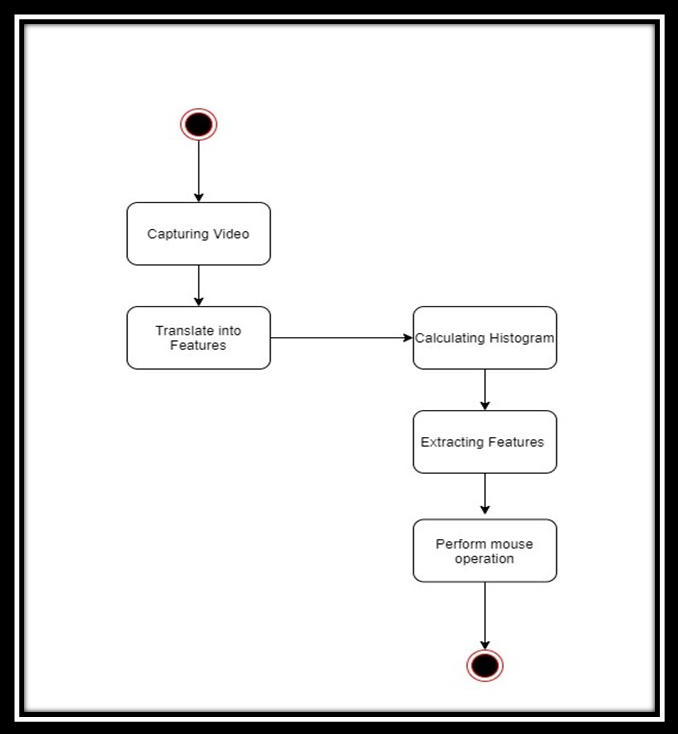
**System Analysis & Design**

4.1 Uses Case Diagram



Shows that the uses case diagram has 3 actors that are that the user the camera and the system. The user is the one that may initialize the system and quit the program at the moment. Besides, the user conjointly go along with the camera is the laptop computer digital camera or Associate in Nursing external digital camera that captures user image in a period of time because the system input. additionally, the utilization case diagram conjointly consists of seven use cases wherever the image acquisition, background subtraction, hand segmentation, options extraction, gesture recognition, and show output square measure the core process stages of the period of time gesture recognition system. the uses cases of the system below is describing the actions that perform by the actors and what will be the expected outcome

**4.2State chart Diagram**



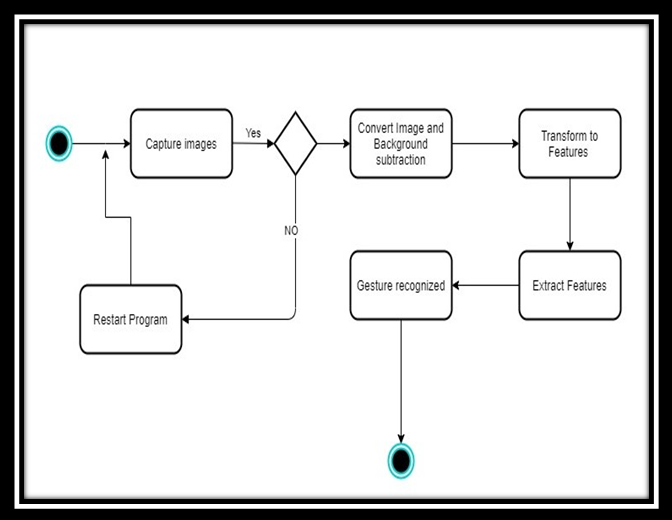
A state chart diagram is one in all the 5 UML diagrams accustomed to model the dynamic nature of a system. They outline different states of the associate object throughout its period and these states square measure modified by events. Statechart diagrams square a measure helpful to model reactive systems. Reactive systems will be outlined as a system that responds to external or internal events. The state chart diagram describes the flow of management from one state to a different state. States square measure is outlined as a condition within which an associate object exists, and it changes once some event is triggered. the foremost vital purpose of the Statechart diagram is to model the period of an associate object from creation to termination. Statechart diagrams are used for forward and reverse engineering of a system. However, the most purpose is to model the reactive system. once structure a state diagram, the square measure many notations and symbols you'll be able to use. With these tools, you'll be able to have a superb presentation and a much better understanding of the system you're attempting to project.

•A State: it's the present physical state of affairs of the associate object. A rounded edge parallelogram sometimes represents it, and within the middle of it's a labeling possibility. you'll be able to conjointly fill it with a color of your selection. The state of the associate object can usually modification thanks to dynamic events.

•Start state: it's the purpose from that a state diagram's flow begins. it is the initial step or stage before any events begin happening. The image is indicated by an obvious circle that you just will fill with any color on E draw easy lay.

•End State: The flow of a state diagram can perpetually come back to associate finish. This end is indicated by the top state, a circle with a spherical border. within the current E draw easy lay version, it seems like an obvious circle once crammed with color.

**4.3 Activity Diagram**



Basic Activity diagram notation and symbols •Start State: a small, filled circle followed by an Associate in Nursing arrow represents the initial action state or the beginning purpose for any activity diagram. For the activity diagram victimization swim lanes, confirm the beginning purpose is placed within the high left corner of the primary column. •Activity or Action state: Associate in Nursing action state represents the non-interruptible action of objects. you'll be able to draw the Associate in Nursing action state in good Draw employing a parallelogram with rounded corners.

• Action flow: Action flows, additionally known as edges and methods, illustrate the transitions from one action state to a different. they're typically drawn with an Associate in Nursing arrowed line.

• Object flow: Object flow refers to the creation and modification of objects by activities. Associate in Nursing object flow arrow from Associate in Nursing action to Associate in Nursing object implies that the action creates or influences the thing. Associate in Nursing object flow arrow from Associate in Nursing object to Associate in Nursing action indicates that the active state uses the thing.

• selections and Branching: A diamond represents a choice with alternate methods. once the Associate in a Nursing activity needs a choice before moving on to the successive activity, add a diamond between the 2 activities. The outgoing alternates ought to be tagged with a condition or guard expression. you'll be able to additionally label one amongst the methods "else."

.

• Guards In UML, guards are an announcement written next to a choice diamond that has to be true before moving next to the successive activity. These don't seem to be essential, however, are helpful once a particular answer, like "Yes, 3 labels are written," is required before moving forward.

• Synchronization

A fork node is employed to separate one incoming flow into multiple coincident flows. it's described as a straight, slightly thicker line in the Associate in a Nursing activity diagram. A be a part of node joins multiple coincident flows back to one outgoing flow. A fork and being a part of the mode used along are typically spoken as synchronization

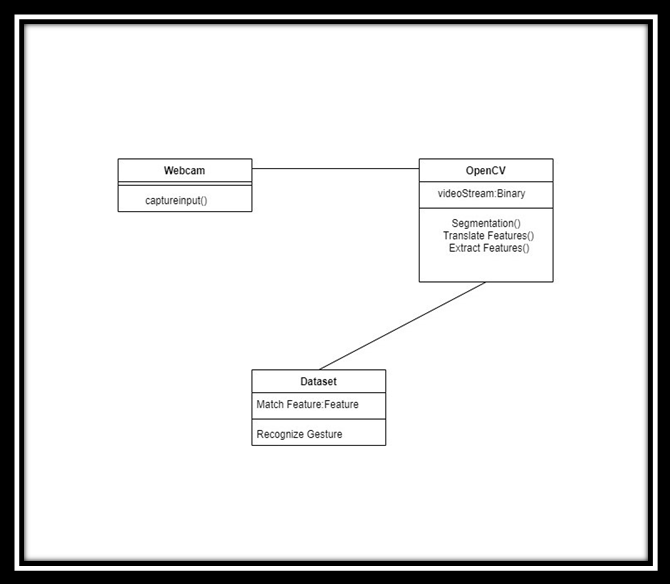
•Sent and Received

Signals represent however activities are often changed from outside the system. they sometimes seem in pairs of sent and received signals, as a result of the state cannot amend till a response is received, very similar to synchronous messages during a sequence diagram. as an example, Associate in the Nursing authorization of payment is required before Associate in Nursing orders are often completed.

• Swim lanes cluster connected activities into one column.

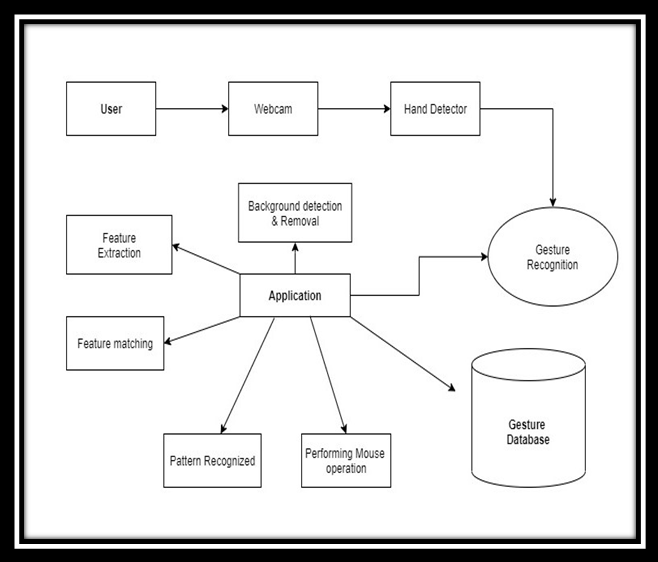
• Final State or finish purpose Associate in Nursing arrow inform to a crammed circle nested within another circle represents the ultimate action state

4.4 Class Diagram



The class diagram could be a graphical notation wont to construct and visualize object-oriented systems. a category diagram within the Unified Modeling Language (UML) could be a sort of static structure diagram that describes the structure of a system by showing the system's: categories their attributes, operations (or methods), and also the relationships among objects.

4.5 Data Flow Diagram



Data flow diagram (DFD) may be a ancient visual illustration of the knowledge flows among a system. A neat and clear DFD will depict the proper quantity of the system demand diagrammatically. It may be manual, automated, or a mix of each.

It shows however knowledge enters and leaves the system, what changes the knowledge, and wherever knowledge is keep.

The objective of a DFD is to indicate the scope and limits of a system as an entire. it should be used as a communication tool between a system analyst and someone World Health Organization plays a neighborhood within the order that acts as a place to begin for redesigning a system. The DFD is additionally known as as an information flow graph or bubble chart.

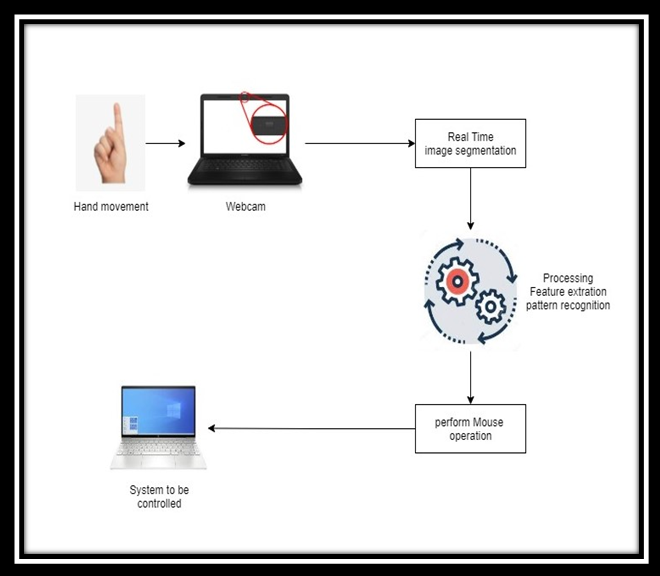
Circle: A circle (bubble) shows a method that transforms knowledge inputs into knowledge outputs.

Data Flow: A flexuous line shows the flow {of knowledge|ofknowledge|of information} into or out of a method or data store.

Data Store: a group of parallel lines shows an area for the gathering of information things. an information store indicates that the information is kept which might be used at a later stage or by the opposite processes in an exceedingly completely different order. the information store will have a component or cluster of components.

Source or Sink: supply or Sink is an associate degree external entity and acts as a supply of system inputs or sink of system outputs.

4.6 Architecture Diagram



An architecture diagram may be a diagram that depicts a system that individuals use to abstract the code system's overall define and build constraints, relations, and limits between elements. It provides a whole read of the physical preparation of the evolution roadmap of the code.

A diagram is comparable to an image. The design diagram examples serve numerous functions. It invariably helps the relevant users to find out concerning system design and apply it within the decision-making procedures. it's crucial to speak info relating to design. However, folks should follow specific steps before creating a diagram for design. These are:

• Breaking down communication barriers

• Reaching a accord

• Decreasing ambiguity

CHAPTER 5

**IMPLEMENTATION**

**5.1 Procedural Description**

The utilizes alternate strategies for control management. Thus, proposes a vision-based indicator system, victimisation hand gestures captured from a digital camera by victimisation media pipe hand detection technique. The goal is to form a system that may recognize the hand gestures-based virtual mouse the computer/laptop in step with those gestures.

We have to follow the procedure: -

1. Open thePyCharm

2. open cv

3. media pipe

4. autopy

5. Run

**5.2Module description**

* **Camera module**:

This module is responsible for connecting and capturing input through the various varieties o image detectors and sends this image to the detection module for process within the kind of frames. The usually used strategies of capturing input square measure knowledge gloves, hand belts, and cameras. In our system, we have a tendency to use the digital camera intrinsic that is price efficient to recognize each static and dynamic gesture. The system has appropriate provision to permit input from a USB primarily based digital camera likewise however this may need some expenditure from the user. The image frames obtained square measure within the kind of a video.

* **Detection module**:

This module is responsible for the image process. The output from the camera module is subjected to completely different image process techniques like color conversion, noise removal, and thresholding following that the image undergoes contour extraction. If the image contains defects, then convexity defects square measure found in keeping with that the gesture is detected. If there are not any defects, then the image is classed using the Haar cascade to detect the gesture. within the case of dynamic gestures, the detection module will be the following; If Microsoft PowerPoint has been launched with a slideshow being enabled and also the digital camera detects palm in movement, for five continuous frames then the dynamic gesture swipe is detected.

* **Interface module**:

This module is responsible for mapping the detected hand gestures to their associated actions. These actions square measure then passed to the acceptable application. The forepart consists of 3 windows. the primary window consists of the video input that's captured from the camera with the corresponding name of the gesture detected. The second window displays the contours found inside the input pictures. The third window displays the sleek threshold version of the image. the advantage of adding the brink and contour window as an area of the Graphical interface is to form the user attentive to the background inconsistencies that might have an effect on the input to the system and so, they'll regulate their portable computer or desktop internet camera to avoid them. this may end in higher performance.

CHAPTER 6

TESTING

6.1 Unit Testing:

In this testing, we have a tendency to tested individual modules against the corresponding specifications set throughout the coming up with part.

Each module was tested for proper logic implementation, and for assurance of meeting the wants, and therefore the needed changes were made

**6.2 Integration Testing:**

Here all the modules, when unit testing, were combined into subsystems then these subsystems were tested to examine the proper integration and conjointly to examine, whether or not the system is fulfilling its necessities.

All the modules were with success tested and correct integration was so ensured.

6.3 System Testing:

• Here all the subsystems were properly integrated into the complete code. then whole code was tested for its correctness and to confirm that it meets all the wants.

• The whole code passed these tests and needed changes and modifications were created and also the code is currently prepared to be used.

6.4 Validation Testing:

• In this testing, we have a tendency to determine if the prevailing system complies with the system necessities and performs the dedicated functions that it's designed besides meeting the goals and wishes of the organization.

• All the dedicated operations expressed within the system demand specification were out there and dealing so meeting the goals of the project.

.

6.5 Performance Testing:

• In this testing, we have a tendency to ensure code applications perform properly below their expected employment.

• It could be a testing technique applied to work out system performance in terms of sensitivity, reactivity, and stability below specific employment.

**CHAPTER 7**

**ALGORITHM AND TECHNIQUES**

For the aim of detection of hand gestures and hand chase, the MediaPipe framework is employed, and the OpenCV library is employed for pc vision. The rule makes use of the machine learning ideas to trace and recognize hand gestures and hand tips.

**7.1 MediaPipe**

MediaPipe is a framework that is employed for applying a very machine learning pipeline and associated it's a Google open-source framework. Because the framework is built using statistical knowledge, the MediaPipe framework will help for cross-platform programming. The MediaPipe framework is multimodal, which means it can handle a wide range of audio and video formats. The MediaPipe framework is employed by the developers to create and analyze systems using graphs, as well as to create systems for application development. The steps involved in a MediaPipe-based system are specified area unit methods out in the pipeline configuration. The pipeline will be able to run on a variety of platforms, allowing for quantifiability on mobile and desktop devices. The MediaPipe framework is made up of three main components: performance analysis, a framework for collecting device knowledge, and a collection of items known as calculators that are reusable. A pipeline may be a graph that consists of made up of elements known as calculators, with each calculator connected by streams through which knowledge packets pass. Developers are prepared to replace or outline custom calculators at any point in the graph when developing their software. A data-flow diagram is created by combining the calculators and streams; the graph is created with MediaPipe, and each node can be a calculator. As a result, streams connect the area unit of the node. For real-time detection and recognition of a hand or palm, the single-shot detector model is used. The MediaPipe employs the single-shot detector concept. First, it's the first training for a palm detection model within the hand detection module because palms are easier to teach. Furthermore, non-most suppression is much more effective on little items like palms or fists, Locating joint or knuckle coordinates inside the hand region is a model of hand landmarks.

**7.2 OpenCV**

Maybe a pc vision library with object identification picture processing methods. OpenCV may be a library of python artificial language, and it is frequently used to create period pc vision applications are developed by the victimization of the pc vision library The OpenCV library is used in image and video. The construct of improving the human-computer interface victimization pc vision is offered inside the planned hand gesture-based virtual mouse system.

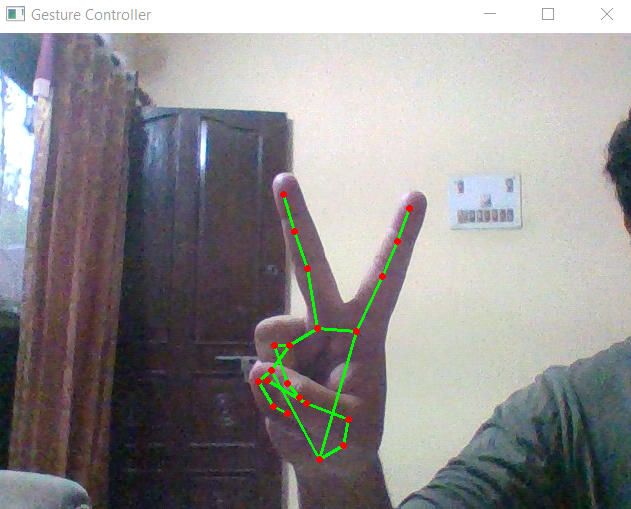


Figure-3 For moving the cursor we will use two fingers.

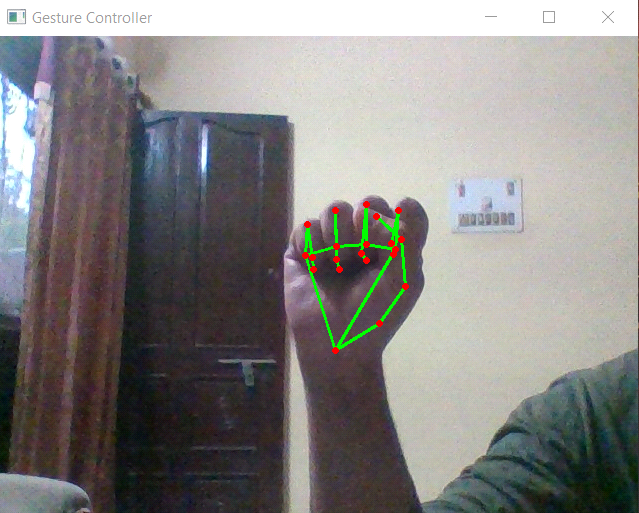


Figure-4 To drag the file we will use all the fingers

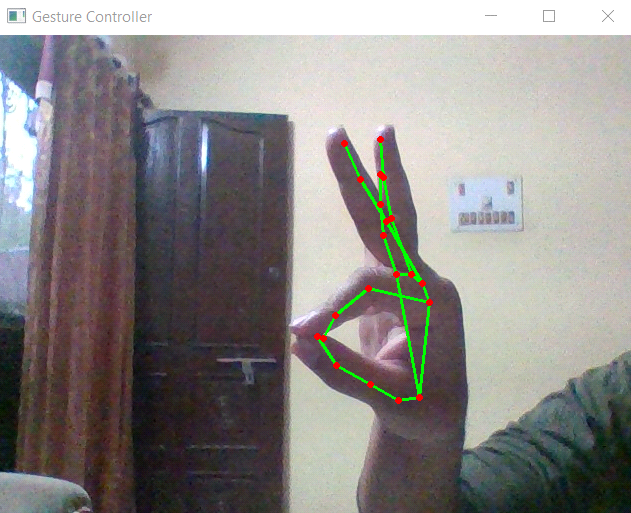


Figure-5 For volume up and down (scroll up and down )we will use one finger and one thumb.

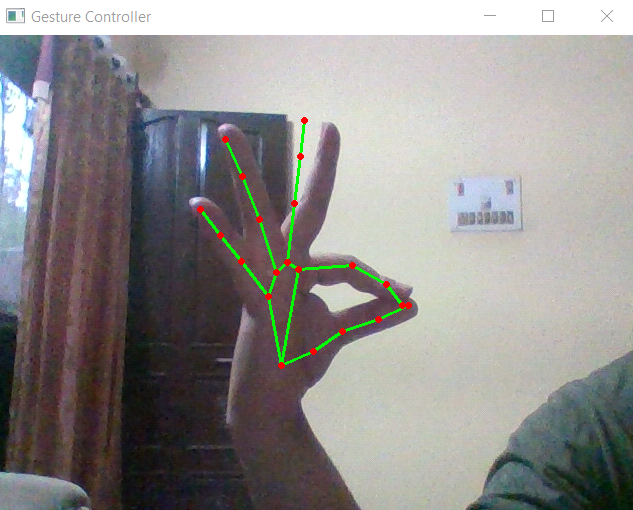


Figure- 6 For brightness up and down we will use one finger and the thumb.

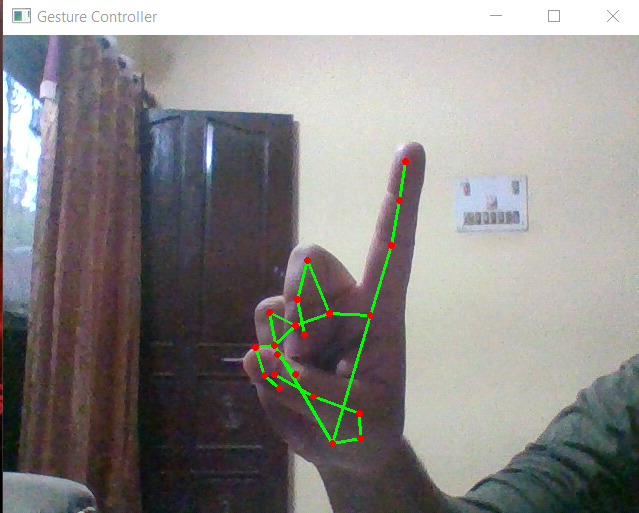


Figure-7 For the left click, we will use one finger.

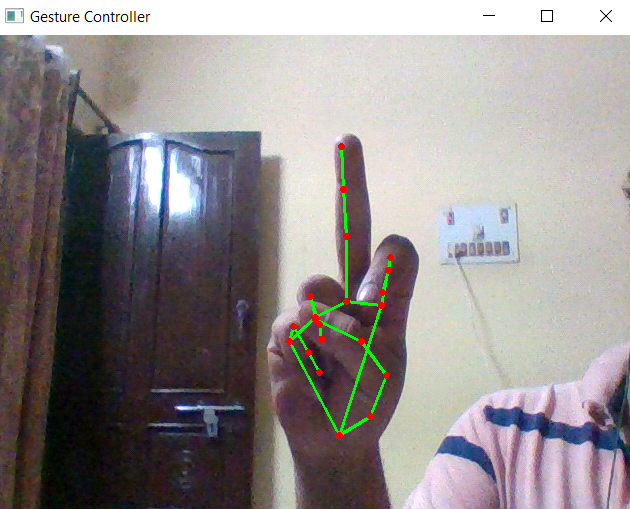


Figure-8 For the right click we will use one finger.

**CHAPTER 8**

**RESULT& DISCUSSION**

In the planned AI virtual mouse system, the concept of advancing the human-computer interaction mistreatment laptop vision is given.

Cross comparison of the testing of the AI virtual mouse system is troublesome as a result of solely restricted numbers of datasets area unit accessible. The hand gestures and fingertip detection are tested in numerous illumination conditions and have additionally been tested with completely different distances from the digital camera for the chase of the hand gesture and hand tip detection. An associate degree experimental take a look at has been conducted to summarize the results shown in Table 1.

The take a look at was performed twenty-five times by four persons leading to 587 gestures with manual labeling, and this take a look at has been created {in completely different in several in numerous} lightweight conditions and at different distances from the screen, and every person tested the AI virtual mouse system ten times in traditional lightweight conditions, five times in faint lightweight conditions, five times in shut distance from the digital camera, and five times in long distance from the digital camera, and therefore the experimental results area unit tabulated in Table 1.

**Table 1: Performance Analysis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hand Gesture | Fingertip Capture | Success | Failure | Accuracy (%) |
| Mouse Movement | 1 | 100 | 0 | 100 |
| Left-Click | 2 | 96 | 4 | 96 |
| Right-Click | 3 | 97 | 2 | 97 |
| Double-click | 4 | 96 | 4 | 96 |
| Scroll up down | 5 | 98 | 2 | 98 |
| No action | 6 | 100 | 0 | 100 |
|  |  | 587 | 12 | 97.8 |

From Table 1, it is often seen that the planned AI virtual mouse system had achieved associate accuracy of regarding 97.8%. From this 97.8 accuracy of the planned AI virtual mouse system, we tend to return to understand that the system has performed well. As seen in Table, the accuracy is low for “Scroll function” as this can be the toughest gesture for the pc to know. The accuracy for scroll performance is low as a result of the gesture used for acting the actual mouse performance is more durable. Also, the accuracy is extremely smart and high for all the opposite gestures. Compared to previous approaches for a virtual mouse, our model worked fine with 97.8 accuracies.

**Table 2: Test Cases -**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case id** | **Scenario** | **Boundary**  **Value** | **Expected Result** | **Actual Result** |
| **1** | Used in a normal environment. | >90% | In a normal environment, hand gestures can be recognized easily. | Hand gestures got easily recognized and work properly. |
| **2** | Used in a bright environment. | >60% | In a brighter environment, software should work fine as it easily detects the hand movements but in brighter conditions, it may not detect the hand gestures as expected. | In bright conditions, the software works very well. |
| **3** | Used in dark environment | <30% | In a dark environment, It should work properly. | In a dark environment, the software didn’t work properly in detecting hand gestures. |
| **4** | Used at a near distance  (15cm) from the webcam. | >80% | At this distance, this software should perform perfectly. | It works fine and all features work properly. |
| **5** | Used at a far distance  (35cm) from the webcam. | >96% | At this distance, this software should work fine. | At this distance, it is working properly. |
| **6** | Used at a farther distance  (60cm) from the webcam. | >60% | At this distance, there will be some problems in detecting hand gestures but it should work fine. | At this distance, The functions of this software work properly. |

**CHAPTER 8**

**CONCLUSIONS AND FUTURE SCOPE**

Due to accuracy(as in table 2) playing a very important role in creating the program as helpful as an actual physical mouse, many techniques had to be enforced. when implanting such a variety of applications there's a huge replacement of the physical mouse i.e., there's no want of any physical mouse. every& each movement of the physical mouse is finished with this motion following the mouse (virtual mouse).

There area unit many options and enhancements required for the program to be a lot of user-friendly, accurate, and versatile in numerous environments. the subsequent describes the enhancements and therefore the options required:

a) good Movement: because of the present recognition method area unit is restricted to a 25cm radius, Associate in Nursing adaptive zoom in/out functions area unit needed to enhance the lined distance, wherever it will mechanically regulate the main focus rate supported the space between the users and therefore the digital camera.

b) higher Accuracy & Performance: The latent period area unit heavily counts on the hardware of the machine, this includes the processing speed of the processor, the dimensions of the accessible RAM, and therefore the accessible options of the digital camera. Therefore, the program might have higher performance once it's running on an honest machine with a digital camera that performs higher in several styles of lighting.

c) Mobile Application: In the future, this net application is conjointly able to use on automaton devices, wherever touch screen idea is replaced by hand gestures.

**REFERENCE**

[1] ’Riza Sande, Neha Marathe, Neha Bhegade, AkankshaLugade, Prof. S. S. Jogdand “virtual mouse using hand gesture”2021International Journal of Advanced Research in Science, Engineering and Technology (IJARSET) pp17124-17132 2021.

[2]A. Agrawal, R. Raj, and S. Porwal, &quot; Vision-based multimodal human-computer interaction using hand and head gestures,&quot; 2013 IEEE Conference on Information &amp; Communication Technologies, Thuckalay, Tamil Nadu, India, 2013, pp. 1288-1292,2013.

[3]HaithamBadi, “Recent methods in vision-based hand gesture recognition”, Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013), Thuckalay, Tamil Nadu, India, India Vol.31, Issue.4, pp.123-141,2013.

[4] S. Thakur, R. Mehra, and B. Prakash, &quot; Vision-based computer mouse control using hand gestures,&quot; 2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI), Faridabad, 2015, pp. 85-87, 2015.

[5] A. Pradhan and B. B. V. L. Deepak, &quot; Obtaining hand gesture parameters using image processing,&quot; 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Chennai, 2015, pp. 168-170, 2015.

[6]S. Shriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay "Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread”2021 Journal of Healthcare Engineering pp 1-9, 2021.

[7]Malavika Suresh, Avigyan Sinha, Aneesh "Real-Time Hand Gesture Recognition Using Deep Learning "2019, International Journal of Innovations &amp; Implementations in Engineering (IJIIE) pp 11-16,2019.

[8]Vijay Kumar Sharma , Vimal Kumar , Md Iqbal , SachineTawara, Vishal Jayaswal “ Virtual Mouse Control using Hand

Class Gesture “ 2020 GIS SCIENCE JOURNAL pp 454-458, 2020.

[9]Abhilash S S, Lisho Thomas, Naveen Wilson 3, Chaithanya C '" VIRTUAL MOUSE USING HAND GESTURE " 2018, International Research Journal of Engineering and Technology (IRJET) pp 3903-3907,2018.

[10] OpenCV Website – [www.opencv.org](http://www.opencv.org/)

[11] Microsoft Research Paper- tp research microsoft.com/en-us/um/people/awf/bmvc02/project.pdf.

# APPENDIX

HGR Software Class Diagram