**REAL TIME AI VIRTUAL MOUSE USING COMPUTER VISION**

**A MINI PROJECT REPORT**

***Submitted by***

**ADITHYA S.S (910619104004)**

**AJAY M (910619104005)**

**ARVINDP (910619104008)**

**HARSHAN R.S (910619104027)**

***In partial fulfilment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**K.L.N. COLLEGE OF ENGINEERING, POTTAPALAYAM**

(An Autonomous Institution, Affiliated to Anna University, Chennai)

****

**ANNA UNIVERSITY: CHENNAI 600 025**

**JULY 2022**

**ANNA UNIVERSITY: CHENNAI 600 025**

**BONAFIDE CERTIFICATE**

Certified that this mini-project report **“REAL TIME AI VIRTUAL MOUSE USING COMPUTER VISION”** is the bonafide work of **“S.S.ADITHYA (910619104004)”, “M.AJAY (910619104005)”**, **“P.ARVIND (910619104008)”, “R.S.HARSHAN (910619104027)”** who carried out the mini-project under my supervision.

|  |  |
| --- | --- |
| **SIGNATURE** | **SIGNATURE** |
| Dr.S.MIRUNA JOE AMALI | Mrs.R.KIRUBAHARI |
| **HEAD OF THE DEPARTMENT** | **SUPERVISOR** |
|  | **ASSISTANT PROFESSOR** |
| Computer science and engineering | Computer science and engineering |
| K.L.N. College of Engineering, | K.L.N. College of Engineering, |
| Pottapalayam, | Pottapalayam, |
| Sivagangai-630 611. | Sivagangai-630 611. |

Submitted for the mini-project viva-voce conducted on

**INTERNAL EXAMINAR EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

The satisfaction and euphoria that accompany the successful completion of any task would be but incomplete without the mention of the people made it possible, whose constant guidance and encouragement crowned our efforts with success.

We extend our gratitude to the Founder, **Late. Thiru. K.L.N. Krishnan, K.L.N College of Engineering and Management Members** for making us march towards the glory of success. We express our sincere thanks to our respected Principal **Dr.A.V. RAM PRASAD**, **M.E, Ph.D., MISTE.FIE,** for all the facilities offered.

We would like to express our deep gratitude and heartfelt thanks to **Dr.S.MIRUNA JOE AMALI, M.E., Ph.D.,** Head of the Department of Computer Science and Engineering, **K.L.N College of Engineering,** who motivated and encouraged us to do this out rival project for this academic year.

Our profound, delightful and sincere thanks to our mini-project Guide and our respected Project Coordinator, **Mrs.R.KIRUBAHARI, M.E.,** whose support was inevitable during the entire period of Our work.

We thank our teaching staffs for sharing their knowledge and view to enhance our project. We also thank our non-teaching staff for extending their technical support to us. I thank my parents for giving me such a wonderful life and my friends for their friendly encouragement throughout the project.

Finally, we thank the Almighty for giving the full health to finish the mini-project successfully.

**ABSTRACT**

The Mouse is one of the wonderful inventions of Human- computer Interaction (HCI) technology. Currently Wireless mouse or a Bluetooth mouse still uses devices and not free of devices completely since it uses a battery for power and a dongle to connect it to the PC. In the proposed AI VIRTUAL MOUSE SYSTEM, this limitation can be overcome by employing webcam or built-in-camera for capturing of hand gestures and hand tip detection using Computer Vision. Based on the hand gestures, the computer can be controlled virtually without the use of physical mouse. Hence, the proposed system will avoid the spread of COVID-19 by eliminating the human dependency of devices to control the computer.

**TABLE OF CONTENTS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | | | **PAGE NO.** |
|  | **ABSTRACT** | | | iv |
|  | **TABLE OF CONTENTS** | | | v |
|  | **LIST OF FIGURES** | | | vii |
|  | **LIST OF ABBREVIATIONS** | | | vii |
| **1** | **INTRODUCTION** | | | **1** |
|  | 1.1 | Purpose | | 4 |
|  | 1.2 | Scope | | 4 |
|  | 1.3 | Applications | | 4 |
|  | 1.4 | Advantages | | 6 |
| **2** | **SYSTEM ANALYSIS** | | | **7** |
|  | 2.1 | Existing System | | 7 |
|  |  | 2.1.1 | Disadvantages | 7 |
|  |  | 2.1.2 | Problem Description | 9 |
|  | 2.2 | Proposed System | | 9 |
|  |  | 2.2.1 | Advantages | 10 |
|  | 2.3 | System Implementation | | 10 |
|  | 2.4 | Description of the Methodology | | 10 |
|  | 2.5 | Specification of Model | | 11 |
| **3** | **SYSTEM SPECIFICATION** | | | **12** |
|  | 3.1 | Hardware and Software Specification | | 12 |
|  |  | 3.1.1 | Hardware Specification | 12 |
|  |  | 3.1.2 | Software Specification | 12 |
|  | 3.2 | Software Description | | 13 |
|  |  | 3.2.1 | Pycharm | 13 |
|  |  | 3.2.2 | Features | 13 |
| **4** | **SYSTEM IMPLEMENTATION** | | | **15** |
|  | 4.1 | List of Modules | | 15 |
|  | 4.2 | Module Descriptions | | 15 |
| **5** | **SYSTEM DESIGN** | | | **17** |
|  | 5.1 | System Architecture | | 17 |
|  | 5.2 | Dataflow Diagram | | 18 |
| **6** | **SOFTWARE DESCRIPTION** | | | **22** |
| **7** | **SYSTEM TESTING** | | | **26** |
|  | 7.1 | System Testing | | 27 |
|  | 7.2 | Blackbox Testing | | 27 |
|  | 7.3 | Integration Testing | | 28 |
| **8** | **SAMPLE CODING** | | | 31 |
| **9** | **SCREENSHOTS** | | | 35 |
| **10** | **CONCLUSION** | | | 41 |
| **11** | **FUTURE ENHANCEMENT** | | | 43 |
| **12** | **REFERENCES** | | | 44 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **TITLE** | **PAGE NO.** |
| 5.1 | System Architecture | 12 |
| 5.2 | Data Flow Diagram | 12 |
| 9.1 | Capturing the Video and Processing | 15 |
| 9.2 | Detecting  Which Finger is Up and Performing the Particular Function | 17 |
| 9.3a | For the Mouse to Perform Left Button Click | 25 |
| 9.3b | 33 |
| 9.4 | For the Mouse to Perform Right Click | 36 |
| 9.5 | For the Mouse to Perform Scroll Up Function | 37 |
| 9.6 | For the Mouse to Perform Scroll Down Function | 38 |
| 9.7 | For No Action to be Performed on Screen | 39 |

**CHAPTER 1**

**INTRODUCTION**

**AI Virtual Mouse**

With the development technologies in the areas of augmented reality and devices that we use in our daily life, these devices are becoming compact in the form of Bluetooth or wireless technologies. The AI virtual mouse system that makes use of the hand gestures and hand tip detection for performing mouse functions in the computer using computer vision. The main objective of the proposed system is to perform computer mouse cursor functions and scroll function using a web camera or a built-in camera in the computer instead of using a traditional mouse device. Hand gesture and hand tip detection by using computer vision is used as a HCI with the computer. With the use of the AI virtual mouse system, we can track the fingertip of the hand gesture by using a built-in camera or web camera and perform the mouse cursor operations and scrolling function and also move the cursor with it. While using a wireless or a Bluetooth mouse, some devices such as the mouse, the dongle to connect to the PC, and also, a battery to power the mouse to operate are used, but in this paper, the user uses his/her built-in camera or a webcam and uses his/her hand gestures to control the computer mouse operations. In the proposed system, the web camera captures and then processes the frames that have been captured and then recognizes the various hand gestures and hand tip gestures and then performs the particular mouse function. Python programming language is used for developing the AI virtual mouse system, and also, OpenCV which is the library for computer vision is used in the AI virtual mouse system. In the proposed AI virtual mouse system, the model makes use of the Media Pipe package for the tracking of the hands and for tracking of the tip of the hands, and also, Pynput, Autopy, and PyAutoGUI packages were used for moving around the window screen of the computer for performing functions such as left click, right click, and scrolling functions. The results of the proposed model showed very high accuracy level, and the proposed model can work very well in real-world application with the use of a CPU without the use of a GPU.

**ARTIFICIAL INTELLIGENCE**

Artificial Intelligence (AI) is the branch of computer sciences that emphasizes the development of intelligence machines, thinking and working like humans. For example, speech recognition, problem-solving, learning and planning.

Today, Artificial Intelligence is a very popular subject that is widely discussed in technology and business circles. Many experts and industry analysts argue that AI or machine learning is the future – but if we look around, we are convinced that it’s not the future – it is the present.

With the advancement in technology, we are already connected to AI in one way or the other – whether it is Siri, Watson or Alexa. Yes, the technology is in its initial phase and more and more companies are investing resources in machine learning, indicating a robust growth in AI products and apps in the near future.

The following statistics will give you an idea of growth!

– In 2014, more than $300 million was invested in AI startups, showing an increase of 300%, compared to the previous year (Bloomberg).

– By 2018, 6 billion connected devices will proactively ask for support. (Gartner)

– By the end of 2018, “customer digital assistants” will recognize customers by face and voice across channels and partners (Gartner).

– Artificial intelligence will replace 16% of American jobs by the end of the decade (Forrester).

– 15% of Apple phone owners’ users use Siri’s voice recognition capabilities (BGR).

Unlike general perception, artificial intelligence is not limited to just IT or technology industry; instead, it is being extensively used in other areas such as medical, business, AI in education, law, and manufacturing.

**1.1 PURPOSE**

The main objective of the proposed AI virtual Mouse system is to develop an alternative to the regular and traditional mouse system to perform and control the mouse functions, and this can be achieved with the help of a web camera that captures the hand gestures and hand tip and then processes these frames to perform the particular mouse function such as left click, right click, and scrolling function.

**1.2 SCOPE**

The AI virtual mouse system is useful for many applications; it can be used to reduce the requirement for using the physical mouse, and it can be used in situations where we cannot use the physical mouse. The system eliminates the usage of devices, and it improves the human-computer interaction.

**1.3 APPLICATIONS**

(i) The proposed model has a greater accuracy of 99% which is far greater than the that of other proposed models for virtual mouse, and it has many applications.

(ii) Amidst the COVID-19 situation, it is not safe to use the devices by touching them because it may result in a possible situation of spread of the virus by touching the devices, so the proposed AI virtual mouse can be used to control the PC mouse functions without using the physical mouse.

(iii) The system can be used to control robots and automation systems without the usage of devices.

(iv) AI virtual mouse can be used to play virtual reality and augmented reality-based games without the wireless or wired mouse devices.

(v) Persons with problems in their hands can use this system to control the mouse functions in the computer.

(vi) In the field of robotics, the proposed system like HCI can be used for controlling robots.

(vii) In designing and architecture, the proposed system can be used for designing virtually for prototyping.

**The major modules in this application are,**

 Webcam Functioning

 Capturing and Processing

 Rectangular region for moving through the window

 Detection and performance

**1.4 ADVANTAGES**

Real Time AI Virtual Mouse using Computer Vision has the following advantages they are stated below,

**1. CONTACT FREE:**

This system is contact free thus eliminating the need for any sort of physical contact thereby reducing the spread of diseases like COVID-19.

**2. PORTABILITY:**

This system can be easily carried around and moved making it adaptable under all circumstances.

**3. HANDLING:**

The system takes in naturally occurring hand gestures as input making the movement of the curser much more intimate.

**4. CLEANER OVERALL SETUP:**

The system requires nothing more than what is already available in a laptop, making the overall setup cleaner and minimal by eliminating the need for dongles, mice and other cables.

**5. DOES NOT REQUIRE POWER SUPPLY:**

The system does not require any additional power supply other than that of the PC/Laptop as it does not have the need to charge.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEMS**

The existing system is being implemented by using a wired or a Bluetooth mouse. These devices use dongle to connect to the PC, and also, a battery to power the mouse to operate. Amidst of the COVID-19 situation, it is not safe to use the devices by touching them. It may result in a possible situation of spread of the virus.

**2.1.1 DISADVANTAGES**

Real Time AI Virtual Mouse using Computer Vision has the following disadvantages they are stated below,

1. **HIGHLY STRAIN RELATED WORK:**

Without the help of automation day to day work is becoming highly straining. A user needs to operate a mouse with his/her hand. Continuous clicking on mouse can strain fingers which can result in health problems. While in home, a user when far away from the tube/fan has to go to the particular switch board for switching on/off the controls.

1. **TROUBLE FOR PHYSICALLY CHALLENGED:**

Handicaps face the major problem in day-to-day life and we all are aware of their scenario. Day to day routine without automation gives rise to many problems for them where they are unsuccessful in achieving the controls of home/computer in time. This can be highly frustrating for them.

1. **TIME CONSUMING**:

Normal execution of our routine activities can be time consuming as compared to same activities performed by gestures.

1. **HEALTH PROBLEMS:**

Constantly working on PC can lead to spine related health issues which are chronic problems and take time to heal. Similarly, constant operation of mouse by hand can strain finger muscles.

**2.1.2 PROBLEM DESCRIPTION**

The proposed AI virtual mouse system can be used to overcome problems in the real world such as situations where there is no space to use a physical mouse and also for the persons who have problems in their hands and are not able to control a physical mouse. Also, amidst of the COVID-19 situation, it is not safe to use the devices by touching them because it may result in a possible situation of spread of the virus by touching the devices, so the proposed AI virtual mouse can be used to overcome these problems since hand gesture and hand Tip detection is used to control the PC mouse functions by using a webcam or a built-in camera.

**2.2. PROPOSED SYSTEMS**

* The Real Time AI Virtual Mouse to develop an alternative to the regular and traditional mouse system to perform and control the mouse functions.
* This can be achieved with the help of a web camera that captures the hand gestures and hand tip.
* Then processes these frames to perform the particular mouse function such as left click, right click, and scrolling function.

**2.2.1 ADVANTAGES**

• The main advantage of using hand gestures is to interact with computer as a non-contact human computer input modality.

• Reduce hardware cost by eliminating use of mouse.

• Convenient for users not comfortable with touchpad.

• The framework may be useful for controlling different types of games and other applications dependent on the controlled through user defined gestures.

**2.3 SYSTEM IMPLEMENTATION**

Collection information, checking devices (like webcam) working properly or not, Collection tape or finger ribbon, which should be fit to the fingers, Import packages like NumPy, OpenCV, pynput.mouse, tkinter, Implement the Open Gesture Operation, Fine Tuning.

**2.4 DESCRIPTION OF THE METHODOLOGY**

The proposed AI virtual mouse system is based on the frames that have been captured by the webcam in a laptop or PC. By using the Python computer vision library OpenCV, the video capture object is created and the web camera will start capturing video. The web camera captures and passes the frames to the AI virtual system.

**CHAPTER 3**

**SYSTEM SPECIFICATION**

**3.1 Hardware and Software Specifications**

**3.1.1 Software Requirements**

Operating System : Windows

Programming Language : Python

Software : PyCharm

Version : Python 3

Packages : Required Python packages

**3.1.2 Hardware Requirements**

Hardware : Intel Pentium

Speed : 1.1 GHz

RAM : 4GB

Hard Disk : 5 GB

Camera : Web camera / Built-in Camera

**3.2 SOFTWARE DESCRIPTION**

**3.2.1 PYCHARM**

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python programming language. It is developed by the Czech company JetBrains (formerly known as IntelliJ). It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.

PyCharm is cross-platform, with Windows, macOS and Linux versions. The Community Edition is released under the Apache License, and there is also an educational version, as well as a Professional Edition with extra features (released under a subscription-funded proprietary license).

**3.2.2 FEATURES:**

* Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes
* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
* Python refactoring: includes rename, extract method, introduce variable, introduce constant, pull up, push down and others.
* Eclipse's PyDev, and the more broadly focused Komodo IDE. Support for web frameworks: Django, web2py and Flask [professional edition only.
* Integrated Python debugger.
* Integrated unit testing, with line-by-line code coverage.
* Google App Engine Python development [professional edition only].
* Version control integration: unified user interface for Mercurial, Git, Subversion, Perforce and CVS with change lists and merge.
* Support for scientific tools like Matplotlib, NumPy and SciPy [professional edition only.
* It competes mainly with a number of other Python-oriented IDEs, including

**CHAPTER 4**

**SYSTEM IMPLEMENTATION**

**4.1 LIST OF MODULES**

* Webcam Functioning
* Capturing and Processing
* Rectangular Region for Moving through the Window
* Detection and Performance

**4.2 MODULES DESCRIPTIONS**

**4.2.1 Webcam Functioning**

* It is based on the frames that have been captured by the webcam in a laptop or PC.
* By using OpenCV, the video capture object is created and the web camera will start capturing video.
* The web camera captures and passes the frames to the AI virtual system.

**4.2.2 Capturing And Processing**

* Uses the webcam where each frame is captured 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.

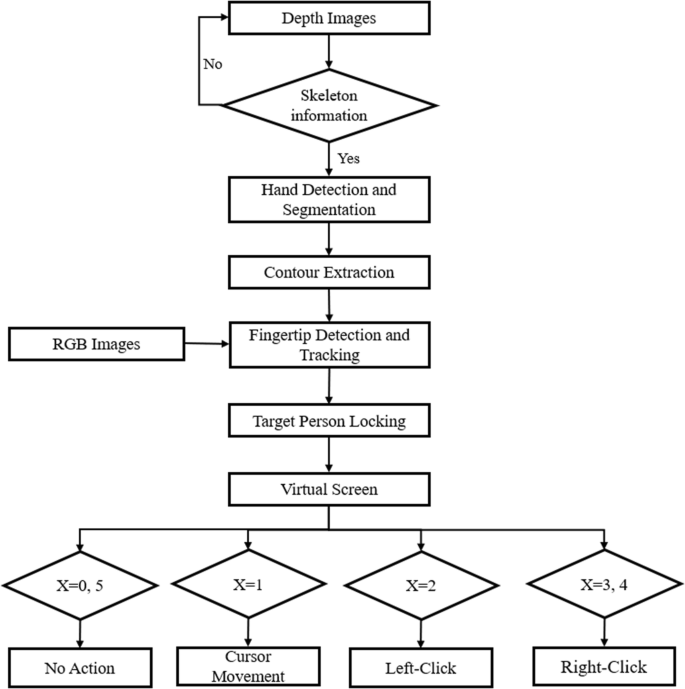
**4.2.3 Creation of Rectangular Region**

* Makes use of the transformational algorithm.
* It converts the co-ordinates for controlling the mouse.
* When hands are detected, a rectangular box is drawn with respect to the computer window in the webcam region.
* We move throughout the window using the mouse cursor.
  + 1. **Detection and Performance**
* We are detecting which finger is up and the respective co-ordinates of the fingers that are up.
* According to that, the particular mouse function is performed.

**CHAPTER 5**

**SYSTEM DESIGN**

**5.1 SYSTEM ARCHITECTURE**



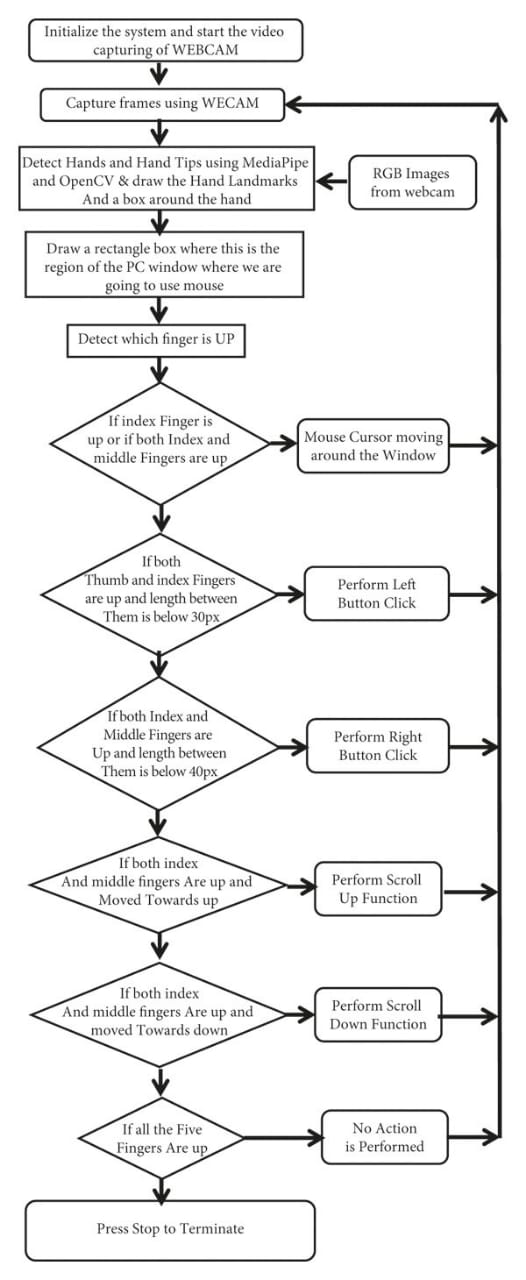
**Figure 5.1: System Architecture of the proposed system**

The proposed system is developed for the college to simplify the allocation of halls and issuing hall tickets to students during exams. It facilitates to access the examination information of a particular student in a particular department. The information is sorted information alphabetically, which will be provided by the teacher for a respective department.

This system is also helpful in finding the examination eligibility criteria of a student in a particular department. Some of the advantages of the proposed system are as follows

* Develop software such that everybody working in exam hall allocation system can handle easily.
* Trainer can store & retrieve data easily. And hence, keeping these major target segments in focus, the system was developed.
* Report can also be provided through print outs.
* Provide a simpler method to store and access information related to exam hall and students.
* Provide a simple interface which will be easily used without much training.
* Reduce paperwork and make all related information accessible easily.

**5.2 DATA FLOW DIAGRAM**



**Figure 5.2: Data Flow Diagram of the proposed system**

* The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.
* The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
* DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
* DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction.

**5.3 MODELING RULES**

There are several common modeling rules when creating DFDs:

* All processes must have at least one data flow in and one data flow out.
* All processes should modify the incoming data, producing new forms of outgoing data.
* Each data store must be involved with at least one data flow.
* Each external entity must be involved with at least one data flow.
* A data flow must be attached to at least one process.

**CHAPTER 6**

**SOFTWARE DESCRIPTION**

**SPECIFICATION:**

* Language Used: Python
* Open CV
* Platform: PyCharm
* Packages: Required Python Packages
* Operating System: Windows 7 and above

**PYCHARM:**

PyCharm is an integrated development environment used in computer programming, specifically for the Python programming language. It is developed by the Czech company JetBrains.

**6.1SOFTWARE COMPONENTS OF PYCHARM**

PyCharm’s huge collection of tools out of the box includes an integrated debugger and test runner; Python profiler; a built-in terminal; integration with major VCS and built-in database tools; remote development capabilities with remote interpreters; an integrated ssh terminal; and integration with Docker and Vagrant.

**6.2 FEATURES OF PYTHON**

**1. Easy to Code:**

Python is a very high-level programming language, yet it is effortless to learn. Anyone can learn to code in Python in just a few hours or a few days. Mastering Python and all its advanced concepts, packages and modules might take some more time. However, learning the basic Python syntax is very easy, as compared to other popular languages like C, C++, and Java.

**2. Easy to Read:**

Python code looks like simple English words. There is no use of semicolons or brackets, and the indentations define the code block. You can tell what the code is supposed to do simply by looking at it.

**3. Free and Open-Source:**

Python is developed under an OSI-approved open-source license. Hence, it is completely free to use, even for commercial purposes. It doesn't cost anything to download Python or to include it in your application. It can also be freely modified and re-distributed. Python can be downloaded from the official Python website.

1. **Robust Standard Library:**

Python has an extensive standard library available for anyone to use. This

means that programmers don’t have to write their code for every single thing unlike other programming languages. There are libraries for image manipulation, databases, unit-testing, expressions and a lot of other functionalities. In addition to the standard library, there is also a growing collection of thousands of components, which are all available in the Python Package Index.

1. **Interpreted:**

When a programming language is interpreted, it means that the source code is executed line by line, and not all at once. Programming languages such as C++ or Java are not interpreted, and hence need to be compiled first to run them. There is no need to compile Python because it is processed at runtime by the interpreter.

**6.3 FEATURES OF OPEN CV:**

OpenCV was originally developed in C++. In addition to it, Python and Java bindings were provided. OpenCV runs on various Operating Systems such as windows, Linux, OSx, FreeBSD, Net BSD, Open BSD, etc.

Using OpenCV library, you can −

* Read and write images
* Capture and save videos
* Process images (filter, transform)
* Perform feature detection
* Detect specific objects such as faces, eyes, cars, in the videos or images.
* Analyse the video, i.e., estimate the motion in it, subtract the background, and track objects in it.

**CHAPTER 7**

**SYSTEM TESTING**

**7.1 SOFTWARE TESTING**

Testing is a process of checking whether the developed system is working according to the original objectives and requirements. It is a set of activities that can be planned in advance and conducted systematically. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the global will be successfully achieved. The best programs are worthless if it produces the correct outputs.

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words, software testing is a verification and validation process.

**7.2 TYPES OF TESTING**

The following three types of testing are used:

* System testing
* Black box testing
* Integration testing

##### **7.2.1 SYSTEM TESTING**

System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

**7.2.2 BLACK BOX TESTING**

Black box testing, also called behavioral testing, focuses on the functional requirements of the software. That is, black testing enables the software engineer to derive sets of input conditions that will fully Exercise all functional requirements fora program. Black box testing is not alternative to white box techniques. Rather it is a complementary approach that is likely to uncover a different class from errors than white box methods. Black box testing attempts to find errors which focuses on inputs, outputs, and principal function of a software module. The starting point of the black box testing is either a specification or code.

##### **7.2.3 INTEGRATION TESTING**

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.

**7.2.4 VERIFICATION AND VALIDATION**

**Verification:**

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it.

**Validation:**

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

#### Basics of software testing:

There are two basics of software testing:

* Integration testing and
* Whitebox testing

##### **Integration testing:**

**Integration Testing** is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated.

**Whitebox Testing:**

White box testing techniques analyze the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing.

**7.3 TEST OBJECTIVES**

[Software Testing](http://istqbexamcertification.com/) has different goals and objectives. The major objectives of Software testing are as follows:

* [Finding defects](http://istqbexamcertification.com/what-is-defect-or-bugs-or-faults-in-software-testing/) which may get created by the programmer while developing the software.
* Gaining confidence in and providing information about the level of [quality](http://istqbexamcertification.com/what-is-software-quality/).
* To prevent defects.
* To make sure that the end result meets the business and user requirements.
* To ensure that it satisfies the BRS that is Business Requirement Specification and SRS that is System Requirement Specifications.
* To gain the confidence of the customers by providing them a quality product.

All the above system testing strategies are carried out in as the development, Documentation and institutionalization of the proposed goals and related policies is essential.

**CHAPTER 8**

**SAMPLE CODING**

import numpy as np

import track\_hand as htm

import time

import autopy

import cv2

wCam, hCam = 1280, 720

frameR = 100

smoothening = 7

pTime = 0

plocX, plocY = 0, 0

clocX, clocY = 0, 0

cap = cv2.VideoCapture(0)

cap.set(3, wCam)

cap.set(4, hCam)

detector = htm.handDetector(maxHands=1)

wScr, hScr = autopy.screen.size()

# print(wScr, hScr)

while True:

# 1. Find hand Landmarks

fingers=[0,0,0,0,0]

success, img = cap.read()

img = detector.findHands(img)

lmList, bbox = detector.findPosition(img)

# 2. Get the tip of the index and middle fingers

if len(lmList) != 0:

x1, y1 = lmList[8][1:]

x2, y2 = lmList[12][1:]

# print(x1, y1, x2, y2)

# 3. Check which fingers are up

fingers = detector.fingersUp()

# print(fingers)

cv2.rectangle(img, (frameR, frameR), (wCam - frameR, hCam - frameR),

(255, 0, 255), 2)

# 4. Only Index Finger : Moving Mode

if fingers[1] == 1 and fingers[2] == 0:

# 5. Convert Coordinates

x3 = np.interp(x1, (frameR, wCam - frameR), (0, wScr))

y3 = np.interp(y1, (frameR, hCam - frameR), (0, hScr))

# 6. Smoothen Values

clocX = plocX + (x3 - plocX) / smoothening

clocY = plocY + (y3 - plocY) / smoothening

# 7. Move Mouse

autopy.mouse.move(wScr - clocX, clocY)

cv2.circle(img, (x1, y1), 15, (255, 0, 255), cv2.FILLED)

plocX, plocY = clocX, clocY

# 8. Both Index and middle fingers are up : Clicking Mode

if fingers[1] == 1 and fingers[2] == 1:

# 9. Find distance between fingers

length, img, lineInfo = detector.findDistance(8, 12, img)

print(length)

# 10. Click mouse if distance short

if length < 40:

cv2.circle(img, (lineInfo[4], lineInfo[5]),

15, (0, 255, 0), cv2.FILLED)

autopy.mouse.click()

# 11. Frame Rate

cTime = time.time()

fps = 1 / (cTime - pTime)

pTime = cTime

cv2.putText(img, str(int(fps)), (20, 50), cv2.FONT\_HERSHEY\_PLAIN, 3,

(255, 0, 0), 3)

# 12. Display

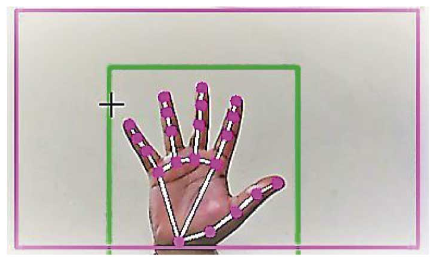
cv2.imshow("Image", img)

cv2.waitKey(1)

**CHAPTER 9**

**SCREENSHOTS**

**9.1 Capturing the Video and Processing**

****

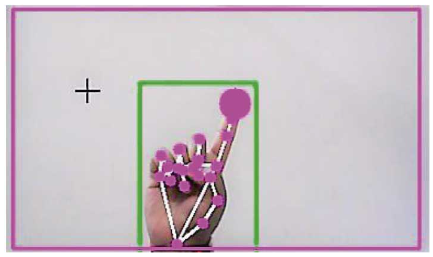
**Figure 9.1 – Capturing the Video and Processing**

The AI virtual mouse system uses the webcam where each frame is captured 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.

**9.2 Rectangular Region for Moving through the Window**

The AI virtual mouse system makes use of the transformational algorithm, and it converts the co-ordinates if fingertip from the webcam screen to the computer window full screen for controlling the mouse. When the hands are detected and when we find which finger is up for performing the specific mouse function, a rectangular box is drawn with respect to the computer window in the webcam region where we move throughout the window using the mouse cursor.

**9.3 Detecting Which Finger Is Up and Performing the Particular Mouse Function**

****

**Figure 9.2 – Detecting Which Finger Is Up and Performing the Particular Mouse Function**

In this stage, we are detecting which finger is up using the tip of the respective finger that we found using the MediaPipe and the respective co-ordinates of the fingers that are up as shown in the figure, and according to that, the particular mouse function is performed.

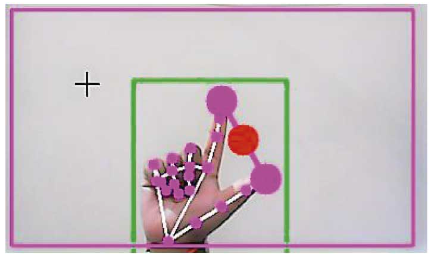
**9.4 Mouse Functions Depending on the Hand Gestures and Hand Tip Detection Using Computer Vision**

**9.4.1 For the Mouse Cursor Moving around the Computer Window**

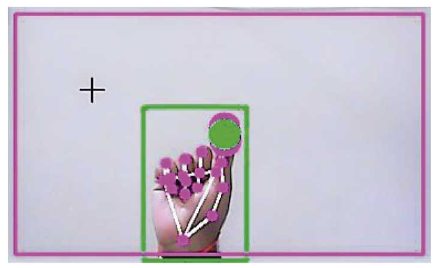
If the index finger is up with tip Id = 1 or both the index finger with tip Id = 1 and the middle finger with tip Id = 2 are up, the mouse cursor is made to move around the window of the computer using the AutoPy package of Python, as shown in the figure.

**9.4.2 For the Mouse to Perform Left Button Click**

In both the index finger with tip Id = 1 and the thumb finger with tip Id = 0 are up and the distance between the two fingers is lesser than 30px, the computer is made to perform the left mouse button click using the pynput Python package, as shown in the figure



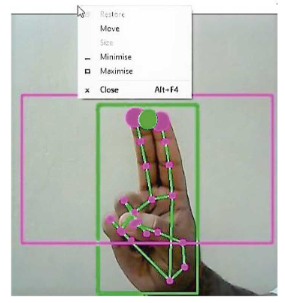
**Figure 9.3a – For the Mouse to Perform Left Button Click**



**Figure 9.3b – For the Mouse to Perform Left Button Click**

**9.4.3 For the Mouse to Perform Right Click**

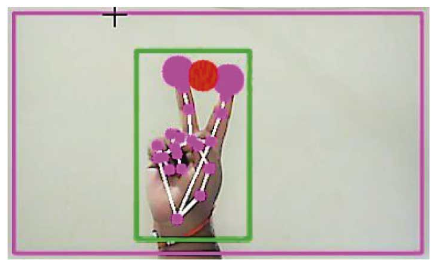
If both the index finger with tip Id = 1 and the middle finger with tip Id = 2 are up and the distance between the two fingers is lesser than 40px, the computer is made to perform the right mouse button click using the punput Python package, as shown in figure.



**Figure 9.4 – For the Mouse to Perform Right Click**

**9.4.4 For the Mouse to Perform Scroll up Function**

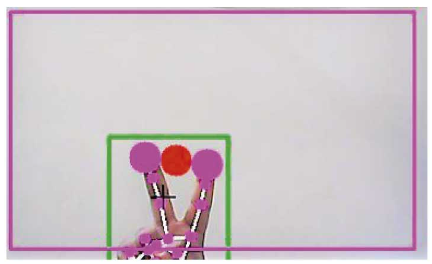
If both the index finger with tip Id = 1 and the middle finger with tip Id = 2 are up and the distance between the two fingers is greater than 40px and if the two fingers are moved up the page, the computer is made to perform the scroll up mouse function using the PyAutoGUI Pyton package, as shown in the figure.



**Figure 9.5 – For the Mouse to Perform Scroll Up Function**

**9.4.5 For the Mouse to Perform Scroll Down Function**

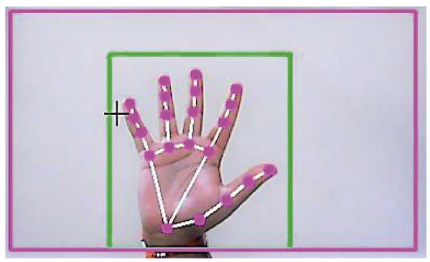
In both the index finger with tip Id = 1 and the middle finger with tip Id = 2 are up and the distance between the two fingers is greater than 40px and if the two fingers are moved down the page, the computer is made to perform the scroll down mouse function using the PyAutoGUI Python package, as shown in the figure.



**Figure 9.6 – For the Mouse to Perform Scroll Down Function**

**9.4.6 For no Action to be Performed on the Screen**

If all the fingers are up with the tip Id = 0, 1, 2, 3, and 4, the computer is made to not perform any mouse events in the screen as shown in the figure.



**Figure 9.7 – For No Action to be Performed on Screen**

**CHAPTER 10**

**CONCLUSION**

From the results of the model, we can come to a conclusion that the proposed AI virtual mouse system has performed very well and has a greater accuracy compared to the existing models and also the model overcomes most of the limitations of the existing systems. Since the proposed model has greater accuracy, the AI virtual mouse can be used for real-world applications, and also, it can be used to reduce the spread of COVID-19, since the proposed mouse system can be used virtually using hand gestures without using the traditional physical mouse.

**Data Availability:**

The hand tracking data used to support the findings of this study are included within the article. The study uses Google’s framework; Hence no new data are needed to train the model.

**CHAPTER 11**

**FUTURE ENHANCEMENT**

The proposed AI virtual mouse has some limitations such as small decrease in accuracy of the right click mouse function and also the model has some difficulties in executing clicking and dragging to select the text. These are some of the limitations of the proposed AI virtual mouse system, and these limitations will overcome in our future work.

Furthermore, the proposed method can be developed to handle the keyboard functionalities along with the mouse functionalities virtually which is another future scope of Human-Computer Interaction (HCI).

**CHAPTER 12**

**REFERENCES**

1. D. L. Quam, “Gesture recognition with a DataGlove,” IEEE Conference on Aerospace and Electronics, vol. 2, pp. 755–760, 1990.
2. L. Thomas, “Virtual mouse using hand gesture,” International Research Journal of Engineering and Technology (IRJET, vol. 5, no. 4, 2018).
3. K. P. Vinay, “Cursor control using hand gestures,” *International Journal of Critical Accounting*, vol. 0975–8887, 2016.
4. S. U. Dudhane, “Cursor control system using hand gesture recognition,” *IJARCCE*, vol. 2, no. 5, 2013.
5. D.-H. Liou, D. Lee, and C.-C. Hsieh, “A real time hand gesture recognition system using motion history image,” in *Proceedings of the 2010 2nd International Conference on Signal Processing Systems*, IEEE, Dalian, China, July 2010.
6. D.-S. Tran, N.-H. Ho, H.-J. Yang, S.-H. Kim, and G. S. Lee, “Real-time virtual mouse system using RGB-D images and fingertip detection,” *Multimedia Tools and Applications Multimedia Tools and Applications*, vol. 80, no. 7, pp. 10473–10490, 2021.
7. K. H. Shibly, S. Kumar Dey, M. A. Islam, and S. Iftekhar Showrav, “Design and development of hand gesture based virtual mouse,” in *Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, pp. 1–5, Dhaka, Bangladesh, May 2019.