MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI

A

Project Report On,

"Virtual Mouse"

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THREE YEARS DIPLOMA IN

Computer Engineering

SUBMITTED BY

Sr. No.	First Name	Middle Name	Surname	Enrollment No.
1	Vishal	Bhimgonda	Desai	2109650128
2	Rakesh	Mahadev	Bandi	2109650125
3	Prajwal	Sadashiv	Chilami	2109650133
4	Rohit	Sanjay	Gorule	2109650126

UNDER THE GUIDANCE OF

MR.R.B.MORE

DEPARTMENT OF COMPUTER ENGINEERING

SANT GAJANAN MAHARAJ RURAL POLYTECHNIC, MAHAGAON

TAL – GADHINGLAJ, DIST – KOLHAPUR – 416503 (MAHARASHTRA)

ACADEMIC YEAR 2023-2024

Maharashtra State Board of
Technical Education
(MSBTE)
Govt. of Maharashtra

This undertaken project work is record of student's own work carried out by them during the academic year 2023-24

SANT GAJANAN MAHARAJ RURAL POLYTECHNIC, MAHAGAON

CERTIFICATE

This is to certify that ***Vishal Bhimgonda Desai (2109650128)*** *has successfully completed and submitted project report in titled* ***"Virtual Mouse"*** *in partial fulfillment for the award of MSBTE, Mumbai three years diploma in the Department of Computer Engineering.*

MR. R. B. MORE
(Guide)

MR. G. K. BIRANGADDI
(Head of Department)

MR. A. T. NARVEKAR
(Project Coordinator)

(External Examiner)

MRS. R. S. PATIL
(Principal)

Seal of
Institut

Maharashtra State Board of
Technical Education
(MSBTE)
Govt. of Maharashtra

This undertaken project work is record of student's own work carried out by them during the academic year 2023-24

SANT GAJANAN MAHARAJ RURAL POLYTECHNIC, MAHAGAON

CERTIFICATE

This is to certify ***Rakesh Mahadev Bandi (2109650125)*** *has successfully completed and submitted project report in titled* ***"Virtual Mouse"*** *in partial fulfillment for the award of MSBTE, Mumbai three years diploma in the Department of Computer Engineering.*

MR. R. B. MORE
(Guide)

MR. G. K. BIRANGADDI
(Head of Department)

MR. A. T. NARVEKAR
(Project Coordinator)

(External Examiner)

MRS. R. S. PATIL
(Principal)

Seal of
Institut

Maharashtra State Board of
Technical Education
(MSBTE)
Govt. of Maharashtra

This undertaken project work is record of student's own work carried out by them during the academic year 2023-24

SANT GAJANAN MAHARAJ RURAL POLYTECHNIC, MAHAGAON

CERTIFICATE

This is to certify that ***Prajwal Sadashiv Chilami (2109650133)*** *has successfully completed and submitted project report in titled* ***"Virtual Mouse"*** *in partial fulfillment for the award of MSBTE, Mumbai three years diploma in the Department of Computer Engineering.*

MR. R. B. MORE
(Guide)

MR. G. K. BIRANGADDI
(Head of Department)

MR. A. T. NARVEKAR
(Project Coordinator)

(External Examiner)

MRS. R. S. PATIL
(Principal)

Seal of
Institut

Maharashtra State Board of
Technical Education
(MSBTE)
Govt. of Maharashtra

This undertaken project work is record of student's own work carried out by them during the academic year 2023-24

SANT GAJANAN MAHARAJ RURAL POLYTECHNIC, MAHAGAON

CERTIFICATE

This is to certify that ***Rohit Sanjay Gorule (2109650126)*** *has successfully completed and submitted project report in titled* ***"Virtual Mouse*** *in partial fulfillment for the award of MSBTE, Mumbai three years diploma in the Department of Computer Engineering.*

-

MR. R. B. MORE
(Guide)

MR. G. K. BIRANGADDI
(Head of Department)

MR. A. T. NARVEKAR
(Project Coordinator)

(External Examiner)

MRS. R. S. PATIL
(Principal)

Seal of
Institut

Sant Gajanan Maharaj Rural Polytechnic, Mahagaon

Institute: Vision, Mission

Vision:

To become globally competitive, technology led organization through all round student development in line with latest requirements in technology, industry and societal.

Mission:

- Inculcating best engineering skills, professional ethics and practices.
- Providing strong foundations by adopting effective teaching learning methods.
- To inculcate best laboratory skills by promoting in house development activities.
- Developing leadership qualities, effective soft skills, critical thinking and attitude of lifelong learning by organizing student centric activities.

Department of Computer Engineering: Vision, Mission

Vision:

To produce technically competent and ethically motivated professionals dedicated to meet industrial and social needs.

Mission:

M1: To provide the students an academic environment essential for technical skill acquisition and intellectual development.
M2: Provide training on latest technologies that meet industry standards.
M2: To practice interdisciplinary culture that upholds values, social commitment and they should grow professionally.

PEO (Program Educational Objectives):

PEO 1. Provide socially responsible, environment friendly solutions to Computer engineering related broad-based problems adapting professional ethics.
PEO 2. Adapt state-of-the-art Computer engineering broad-based technologies to work in multidisciplinary work environments through continuous learning.
PEO3. Solve broad-based problems individually and as a team member communicating effectively in the world of work.

PSO (Program Specific Outcomes):
PSO1. Computer Software and Hardware Usage: Use state-of-the-art technologies for operation and application of computer software and hardware.
PSO2. Computer Engineering Maintenance: Maintain computer engineering related software and hardware systems.

ACKNOWLEDGMENT

We would like to express our gratitude words to our respected guide MR.R.B.MORE sir for his constant encouragement and valuable guidance during the completion of this project. We are very thankful to MR. A. T. Narvekar (Project Coordinator), MR. G. K. Birangaddi sir (HOD) and all the faculty members for those valuable co-operation & guidance during this project work. We thank our MRS. R.S.Patil (Principal) madam for their guidance and direction towards our final year project. We take this opportunity to thank the entire staff members for their co-operation and their helpfulness during this projectwork. Last but not the least assistance offered by various friends and colleagues related directly or indirectly to this work is also gratefully acknowledged.

ABSTRACT

In this project, we design a human computer interaction system using hand gestures. Hand gesture recognition (HGR) is a natural way of Human Machine Interaction and has been applied on different areas. Here we explore the use of hand gestures as a means of human-computer interactions for computer applications. Our goal is to make a system that will recognize a few sets of hand gestures and correspondingly realize the robust control of mouse cursor and mouse clicks with a higher accuracy of gesture recognition.

More specifically it is the use of a convolution neural network (CNN) to recognize the hand gestures in even the harshest lightning/background conditions using only a simple webcam. We will try to adjust the cursor control to the relative movement of our specific gesture set for mouse movement and different mouse events like clicks to different gestures. The final system will be highly extendable and can be further implemented in advanced technologies like augmented reality (AR), virtual reality (VR) for easy and hassle-free navigation and control without the need for keyboard/mouse for computers and controllers for VR/AR headsets.

UNDERTAKING

We hereby confirm that, the work which is being presented in the third year Project Report entitled **"Virtual mouse",** in the fulfillment of requirements for the award of the **Third Year of Engineering in Computer.** and submitted to the Department of Computer Engineering of Sant Gajanan Maharaj Rural Polytechnic Mahagaon, Maharashtra is an authentic record of our work carried out during a period from **July 2023 to April 2024 (6^{th} semester)** under the supervision of **Mr. A. T. Narvekar , Project Coordinator of Computer Department.**

Signature of Students

MR.VISHAL BHIMGONDA DESAI ----------

MR.RAKESH MAHADEV BANDI ----------

MR.PRAJWAL SADASHIV CHILAMI ----------

MR.ROHIT SANJAY GORULE ----------

INDEX

CHAPTER1:
INTRODUCTION & BACKGROUND

1.1 Project Preface –

mouse is the mouse is a hardware input device and there can be some problems like mouse click not functioning properly and etc., as the mouse is a hardware device like any other physical object even the mouse will have a durability time within which is functional and after its durability time we have to change the mouse. As the technology increase everything becomes virtualized such as speech recognition. Speech Recognition is used for recognition and translation of the spoken language into text. Thus, Speech Recognition can replace keyboards in the future, Similarly Eye Tracking which is used to control the mouse pointer with the help of our eye. Eye Tracking can replace mouse in the future.

1.2 Modules :-

1. OPEN
2. CLICK ON TRACK MOUSE BUTTON
3. NO ACTION PERFORMED
4. CURSOR MOVING
5. LEFT BUTTON CLICK
6. RIGHT BUTTON CLICK
7. SCROLLING
8. DRAG AND DROP
9. DOUBLE CLICK

1.3 Objective

This project aims to eliminate the needs of having a physical mouse while able to interact with the computer system through webcam by using various image processing techniques. Other than that, this project aims to develop a Virtual Mouse application that can be operational on all kind of surfaces and environment.

.

The following describes the overall objectives of this project:

- **To design to operate with the help of a webcam.**
  The Virtual Mouse application will be operational with the help of awebcam, as the webcam are responsible to capture the images in real time. The application would not work if there were no webcam detected.
- **To design a virtual input that can operate on all surfaces.**
  The Virtual Mouse application will be operational on all surface and indoor environment, as long the users are facing the webcam while doing the motion gesture.
- **To program the camera to continuously capturing the images, which the images will be analyzed, by using various image processing techniques.**
  As stated above, the Virtual Mouse application will be continuously capturing the images in real time, where the images will be undergoing a series of process, this includes HSV conversion, Binary Image conversion, salt and pepper noise filtering, and more.
- **To convert hand gesture/motion into mouse input that will be set to a particular screen position**.
  The Virtual Mouse application will be programmed to detect the position of the defined colors where it will be set as the position of the mouse pointers. Furthermore, a combination of different colors may result in triggering different types of mouse events, such as the right/left clicks, scroll up/down, and more.

1.4 Purpose

The purpose of this project is to develop a Virtual Mouse application that targets a few aspects of significant development. As the technology increase everything becomes virtualized such as speech recognition. Speech Recognition is used for recognition and translation of the spoken language into text. Thus, Speech Recognition can replace keyboards in the future, Similarly Eye Tracking which is used to control the mouse pointer with the help of our eye. Eye Tracking can replace mouse in the future. Gestures can be in any form like hand image or pixel image, or any human given pose that require less computational difficulty or power for making the devices required for the recognitions to make work. Different techniques are being proposed by the companies for gaining necessary information/data for recognition handmade gestures recognition models. Some models work with special devices such as data glove devices and color caps to develop complex information about gesture provided by the user/human

.

CHAPTER 2:
ANALYSIS

2.1 Module Description

1. OPEN:
   When run the .py file :GUI will be open.

2. CLICK ON TRACK MOUSE BUTTON
   Webcam will be open to capture hand gestures.

3. NO ACTION PERFORMED
   When all the five fingers up then the cursor will stop to moving

4. CURSOR MOVING
   For this action we have to use the index finger.

5. LEFT BUTTON CLICK
   Lower the index finger and raise the middle finger.

6. RIGHT BUTTON CLICK
   For this action first we have to show thumb and middle finger then use the index finger.

7. SCROLLING
   Scroll down: for this action we have to make fiest and show the thumb.
   Scroll up: for this action we have to make fiest and show the middle finger.

8. DRAG AND DROP
   we use the last finger for the selection of the content and then by using index finger move the content.

9. DOUBLE CLICK
   Join/closed both index finger and middle finger then double click action perform.

2.2 Requirement Analysis

Hardware Requirement:

1. Hard disk : 500GB and above
2. RAM : 8GB and above
3. Processor : i5

Software Requirements:

1. VSCODE EDITOR.
2. OPEN-CV LIBRARY.
3. NUMPY LIBRARY.
4. MEDIAPIPE LIBRARY
5. AUTOPY LIBRARY

CHAPTER 3:
SYSTEM DESIGN

3.1 Flow chart

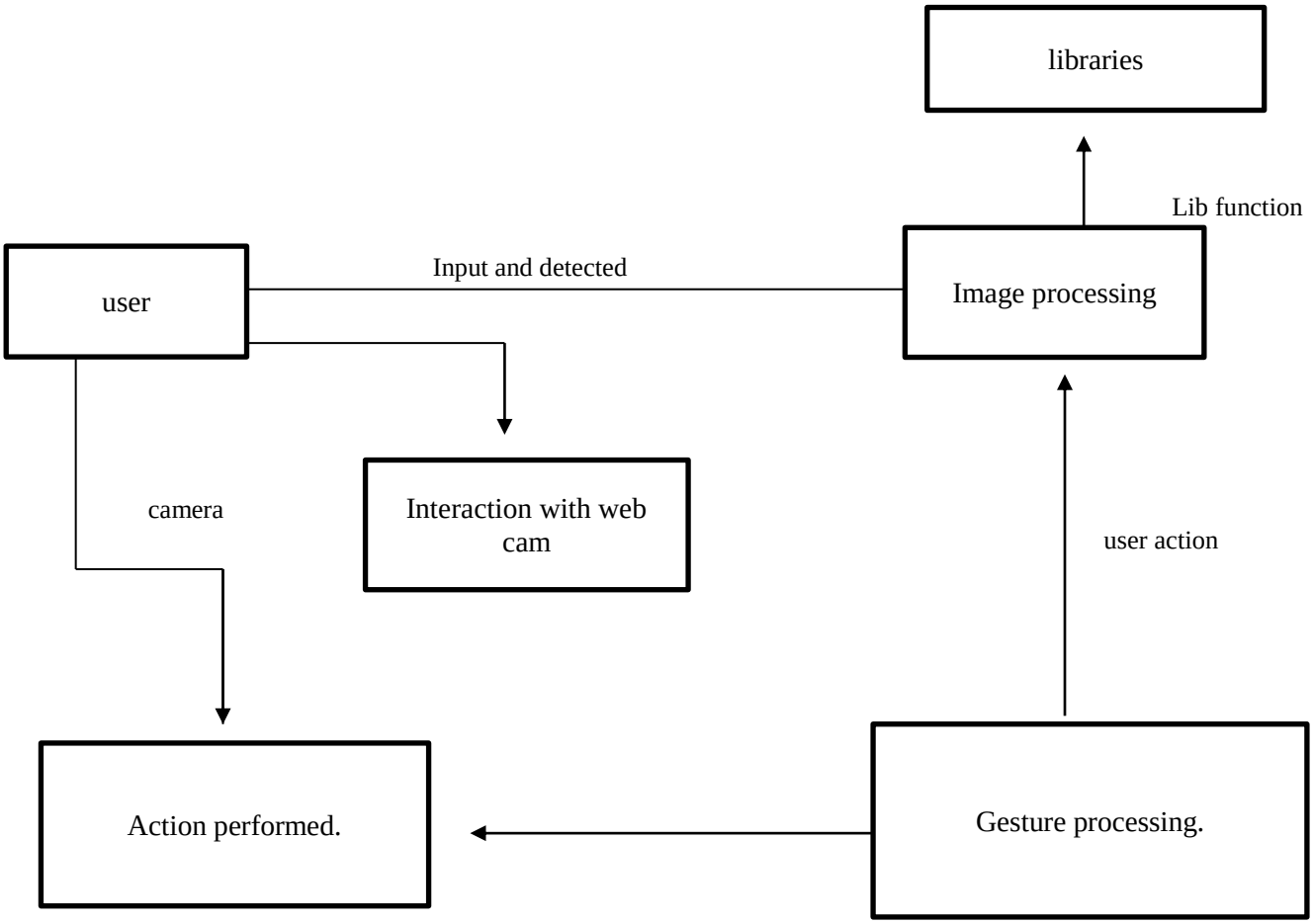


Fig : flow chart

3.2 System block diagram :

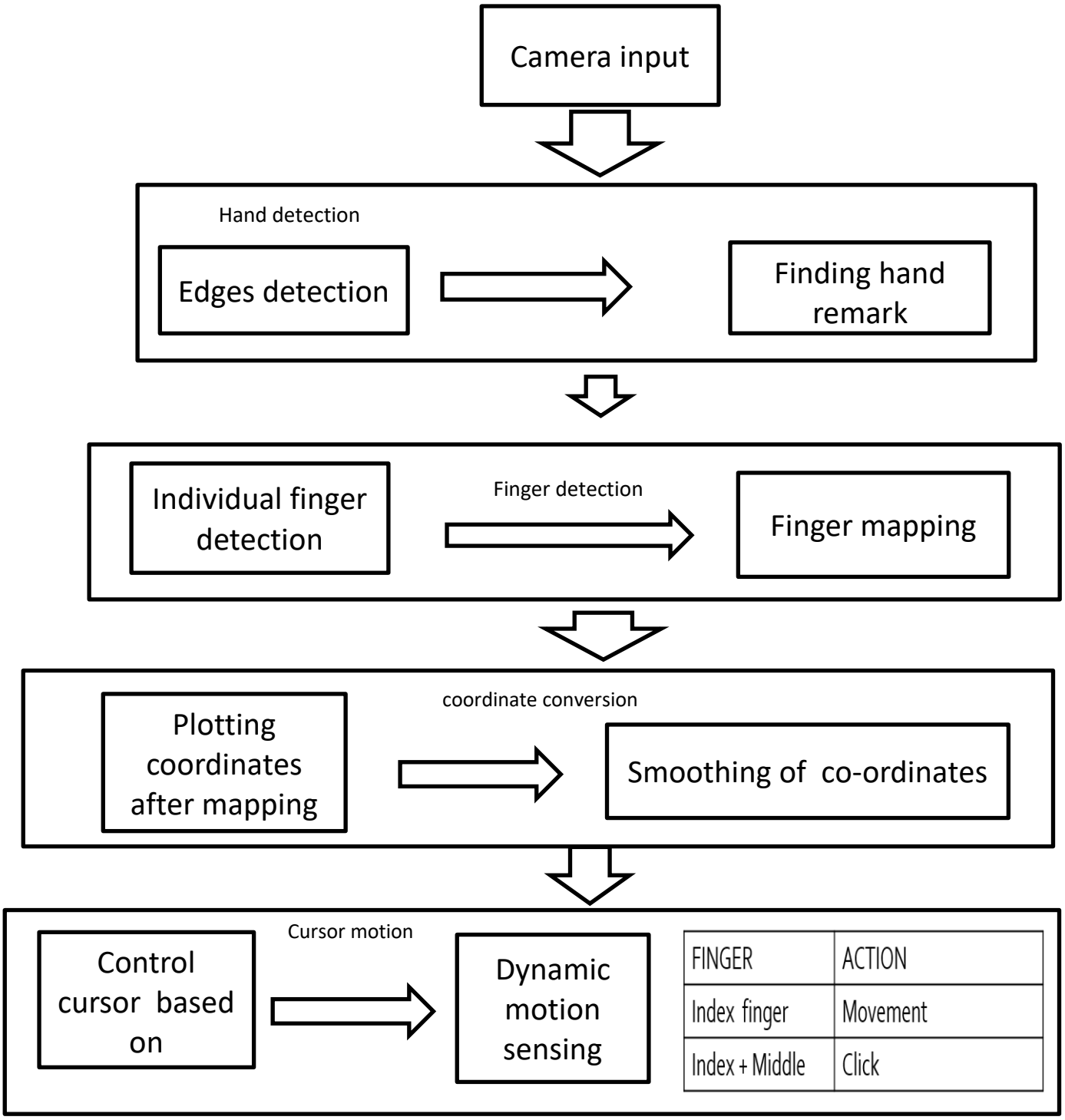


Fig: block diagram

CHAPTER 4 : SYSTEM MODELING

3.3 CLASS DIAGRAM

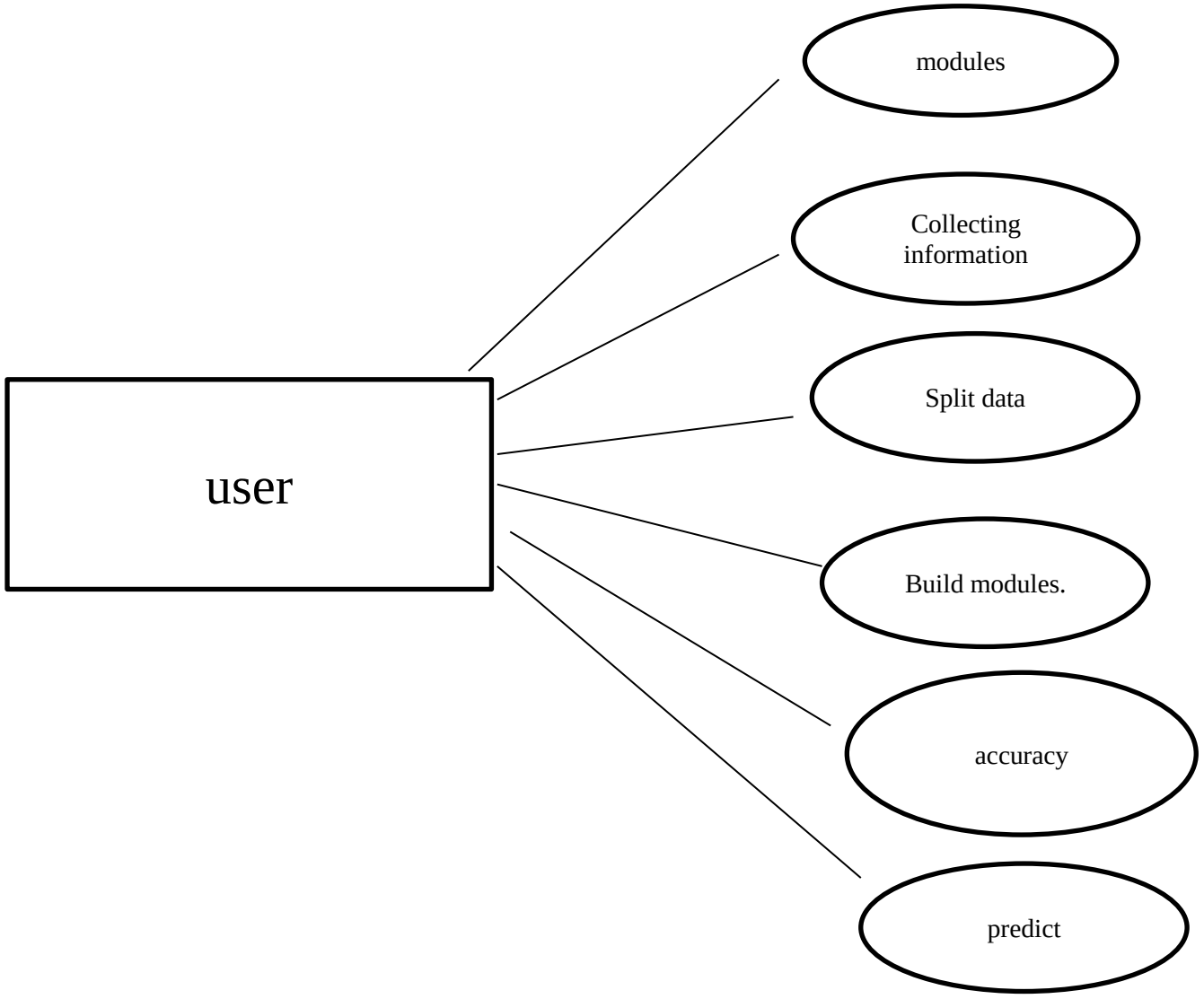


Fig : class diagram

3.3 Architecture:

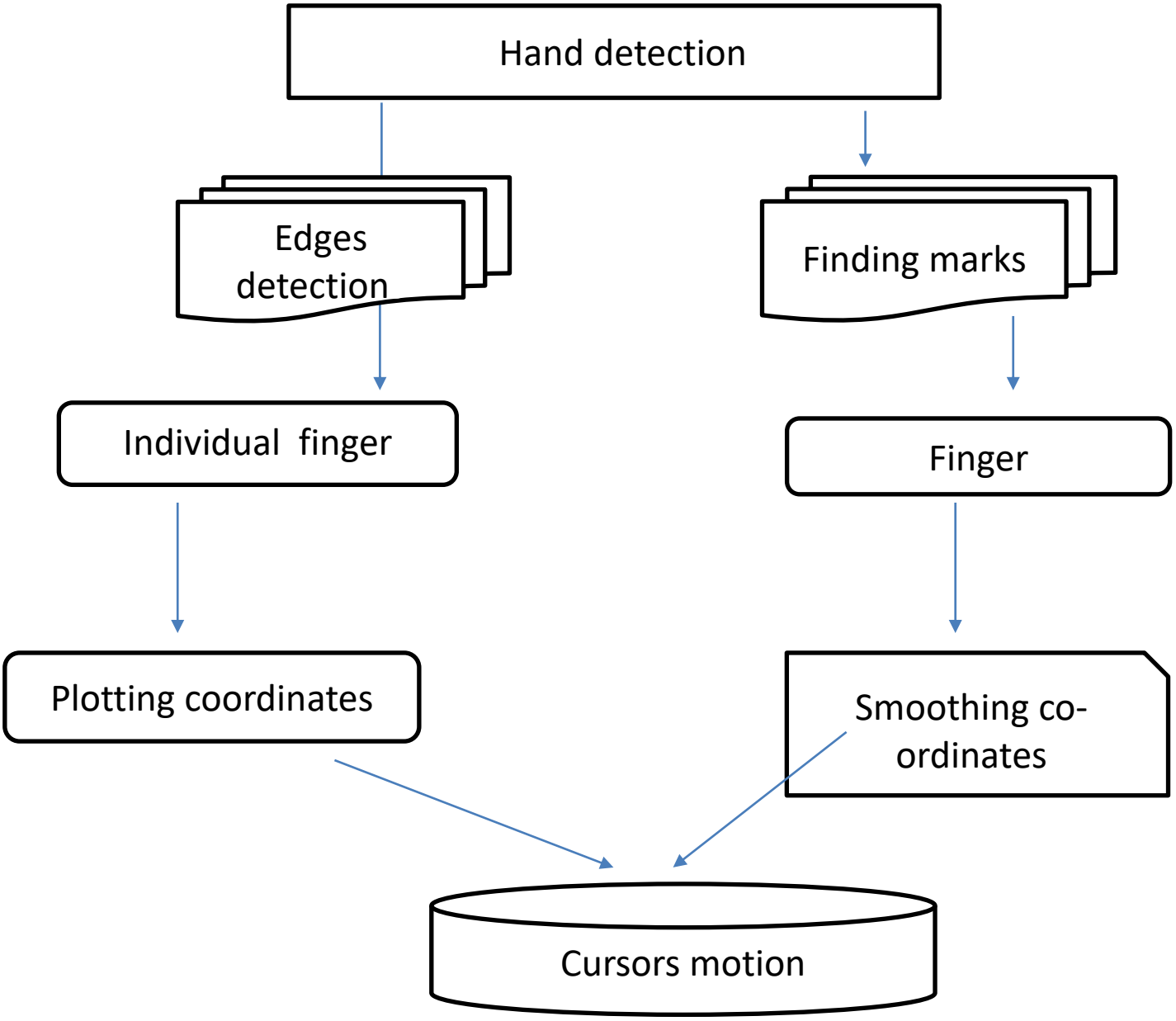


Fig: architecture

3.4 Data flow diagram :

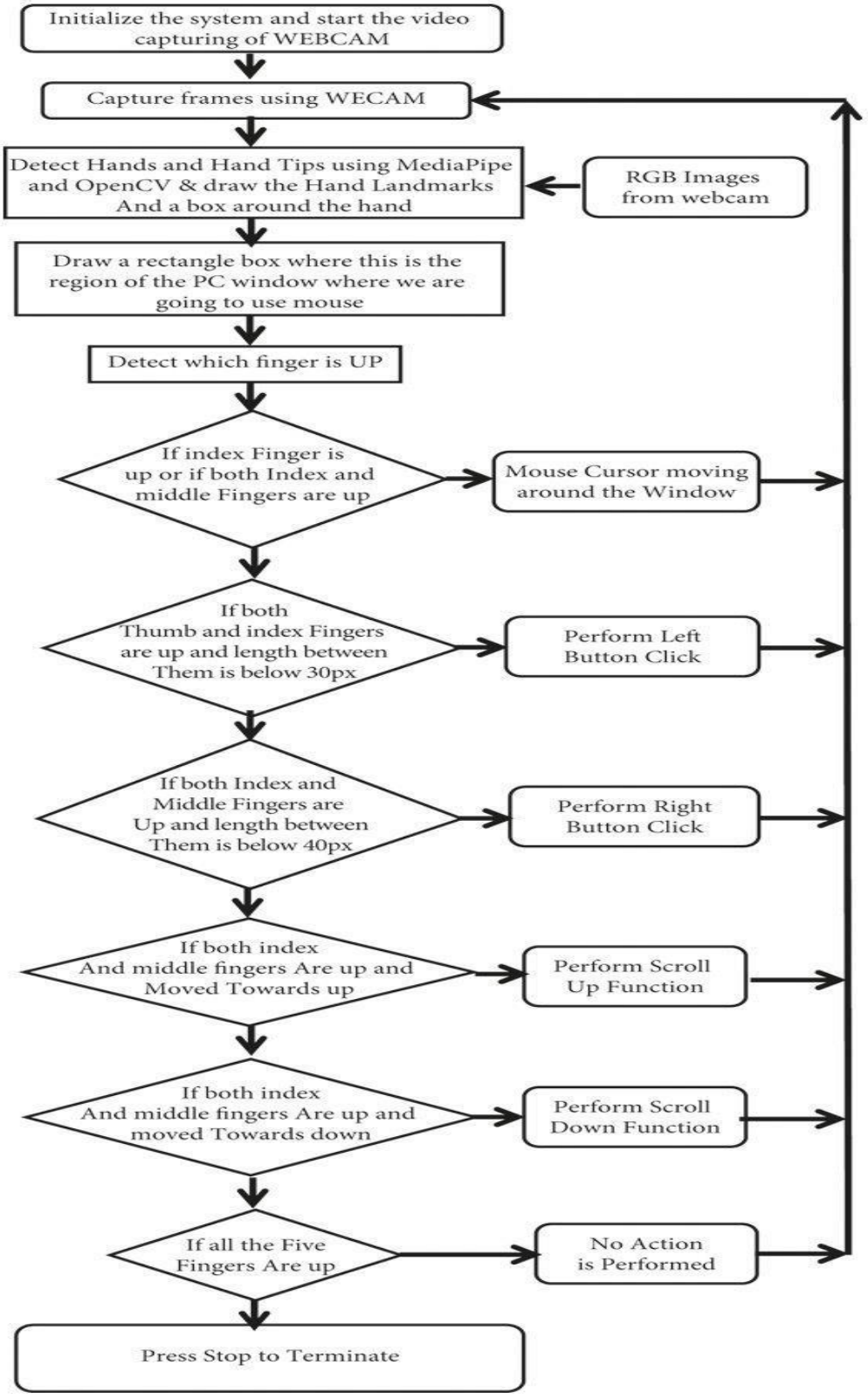


Fig: Data flow diagram

CHAPTER 5
IMPLEMENTATION

4.1 Source Code-

Handtracking.py

```
import cv2
import mediapipe as mp
import time
import math
import numpy as np
class handDetector():    def __init__(self, mode=False, maxHands=2, detectionCon=False,
trackCon=0.5):
        self.mode = mode
        self.maxHands = maxHands
        self.detectionCon = detectionCon
        self.trackCon = trackCon
        self.mpHands = mp.solutions.hands
        self.hands = self.mpHands.Hands(self.mode, self.maxHands,
                                        self.detectionCon, self.trackCon)
        self.mpDraw = mp.solutions.drawing_utils
        self.tipIds = [4, 8, 12, 16, 20]
    def findHands(self, img, draw=True):
        imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        self.results = self.hands.process(imgRGB)
        if self.results.multi_hand_landmarks:
            for handLms in self.results.multi_hand_landmarks:
                if draw:
   self.mpDraw.draw_landmarks(img, handLms,
    self.mpHands.HAND_CONNECTIONS)
        return img
    def findPosition(self, img, handNo=0, draw=True):
        xList = []
        yList = []
        bbox = []
        self.lmList = []
        if self.results.multi_hand_landmarks:
            myHand = self.results.multi_hand_landmarks[handNo]
```

```
for id, lm in enumerate(myHand.landmark):
            h, w, c = img.shape
            cx, cy = int(lm.x * w), int(lm.y * h)
            xList.append(cx)
    yList.append(cy)
            self.lmList.append([id, cx, cy])
            if draw:
                cv2.circle(img, (cx, cy), 5, (255, 0, 255), cv2.FILLED)
        xmin, xmax = min(xList), max(xList)
        ymin, ymax = min(yList), max(yList)
        bbox = xmin, ymin, xmax, ymax
        if draw:
            cv2.rectangle(img, (xmin - 20, ymin - 20), (xmax + 20, ymax + 20),
                          (0, 255, 0), 2)
    return self.lmList, bbox
def fingersUp(self):
    fingers = []            if self.lmList[self.tipIds[0]][1] > self.lmList[self.tipIds[0] - 1][1]:
        fingers.append(1)
    else: append(0)
            for id in range(1, 5):
        if self.lmList[self.tipIds[id]][2] < self.lmList[self.tipIds[id] - 2][2]:
            fingers.append(1)
        else:           fingers.append(0)
         return fingers
def findDistance(self, p1, p2, img, draw=True,r=15, t=3):
    x1, y1 = self.lmList[p1][1:]
    x2, y2 = self.lmList[p2][1:]
    cx, cy = (x1 + x2) // 2, (y1 + y2) // 2
    if draw:        cv2.line(img, (x1, y1), (x2, y2), (255, 0, 255), t)
        cv2.circle(img, (x1, y1), r, (255, 0, 255), cv2.FILLED)
        cv2.circle(img, (x2, y2), r, (255, 0, 255), cv2.FILLED)
        cv2.circle(img, (cx, cy), r, (0, 0, 255), cv2.FILLED)
    length = math.hypot(x2 - x1, y2 - y1)
    return length, img, [x1, y1, x2, y2, cx, cy]
```

```
def main():
    pTime = 0
    cTime = 0
    cap = cv2.VideoCapture(1)
    detector = handDetector()
    while True:
        success, img = cap.read()
        img = detector.findHands(img)
        lmList, bbox = detector.findPosition(img)
        if len(lmList) != 0:
            print(lmList[4])
        cTime = time.time()
        fps = 1 / (cTime - pTime)
        pTime = cTime
        cv2.putText(img, str(int(fps)), (10, 70), cv2.FONT_HERSHEY_PLAIN, 3,
                    (255, 0, 255), 3)
        cv2.imshow("Image", img)
        cv2.waitKey(1)
if __name__ == "__main__":
    main()
```

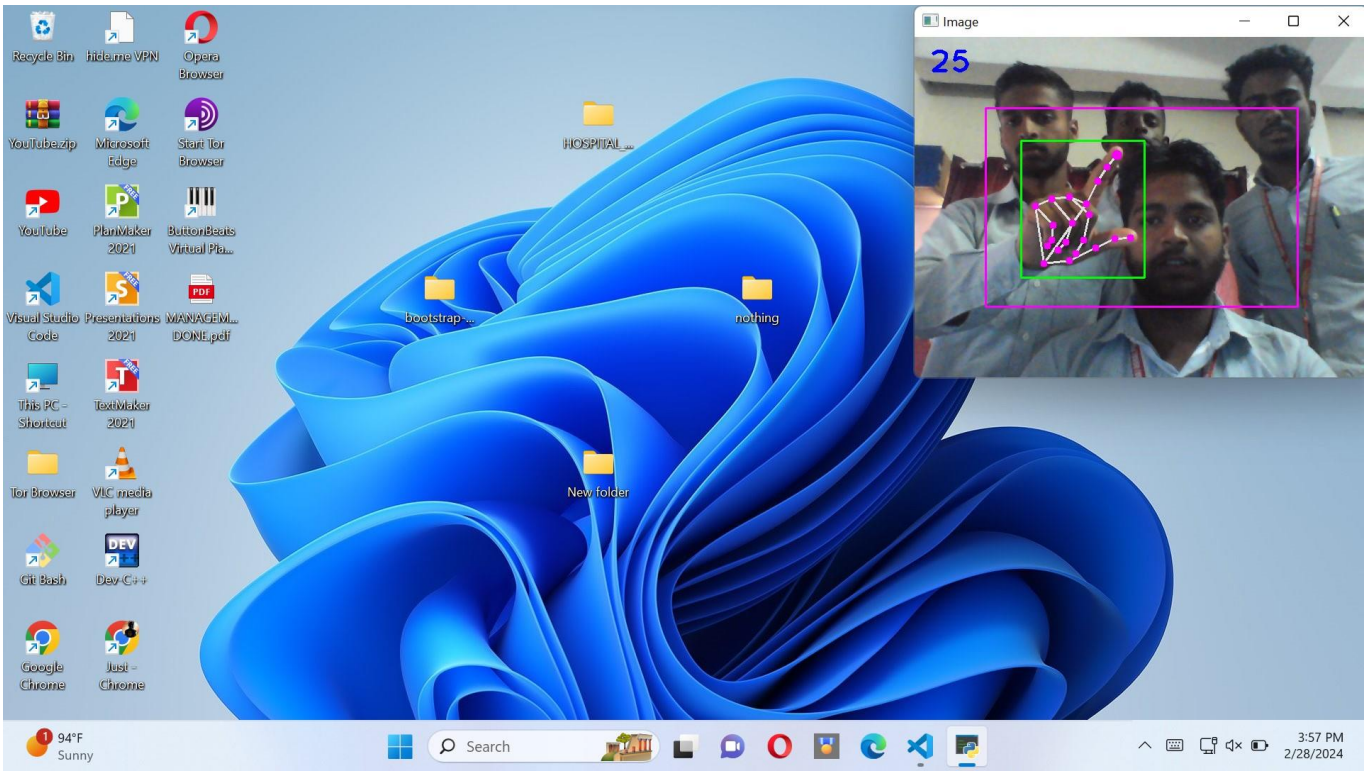
NewMouse.py

```
import cv2
import numpy as np
import time
import HandTracking as ht
import autopy
pTime = 0
width, height = 640, 480
frameR = 100
smoothening = 8
prev_x, prev_y = 0, 0
curr_x, curr_y = 0, 0
cap = cv2.VideoCapture(0)
cap.set(3, width)
cap.set(4, height)
detector = ht.handDetector(maxHands=1)
screen_width, screen_height = autopy.screen.size()
while True:
    success, img = cap.read()
    img = detector.findHands(img)
    lmlist, bbox = detector.findPosition(img)
    if len(lmlist) != 0:
        x1, y1 = lmlist[8][1:]
        x2, y2 = lmlist[12][1:]
        fingers = detector.fingersUp()
        cv2.rectangle(img, (frameR, frameR), (width - frameR, height - frameR), (255, 0, 255), 2)
        if fingers[1] == 1 and fingers[2] == 0:
            x3 = np.interp(x1, (frameR, width - frameR), (0, screen_width))
            y3 = np.interp(y1, (frameR, height - frameR), (0, screen_height))
            curr_x = prev_x + (x3 - prev_x) / smoothening
            curr_y = prev_y + (y3 - prev_y) / smoothening
            autopy.mouse.move(screen_width - curr_x, curr_y)
            cv2.circle(img, (x1, y1), 7, (255, 0, 255), cv2.FILLED)
            prev_x, prev_y = curr_x, curr_y
        elif fingers[0] == 1 and fingers[2] == 1:
```

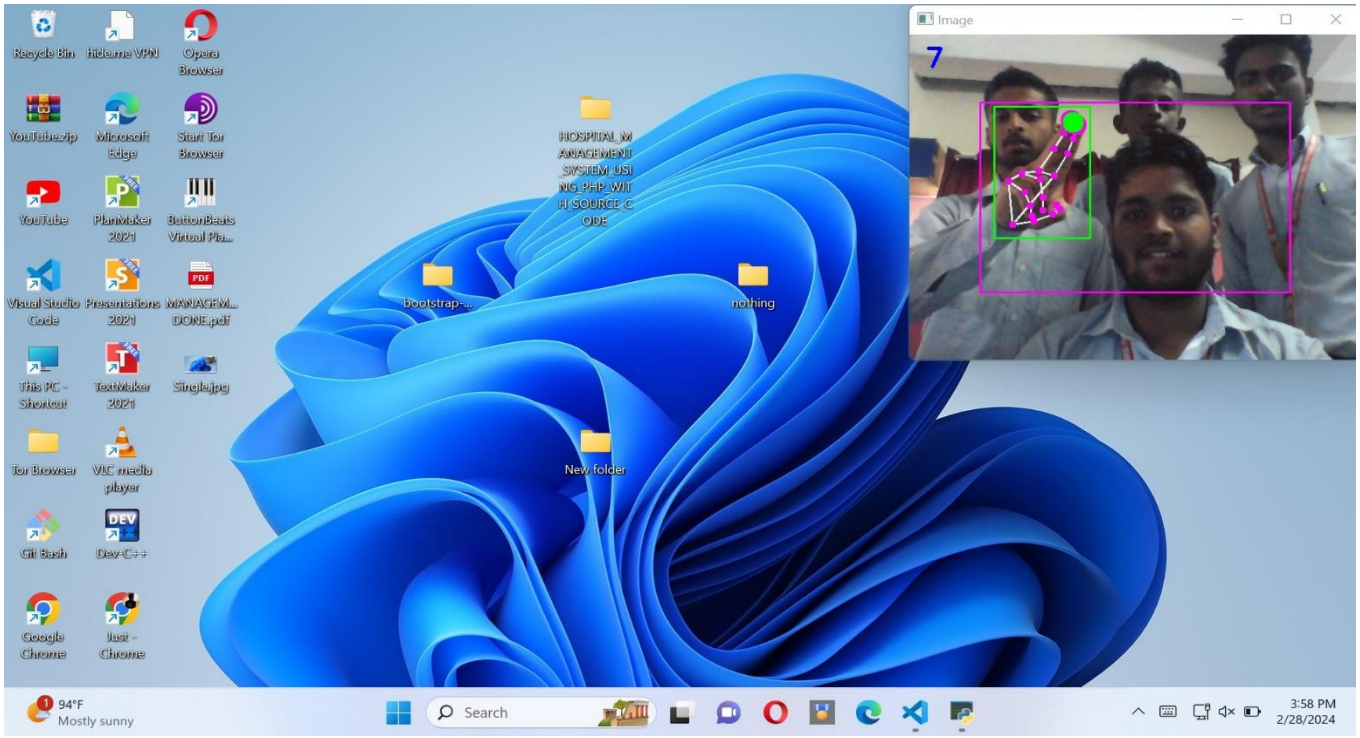
```
autopy.mouse.click(autopy.mouse.Button.RIGHT)
        elif fingers[1] == 1 and fingers[2] == 1:
            length, img, lineInfo = detector.findDistance(8, 12, img)
            if length < 70:
                cv2.circle(img, (lineInfo[4], lineInfo[5]), 15, (0, 255, 0), cv2.FILLED)
                autopy.mouse.click()


        elif fingers[0] == 0 and fingers[1] == 0 and fingers[2] == 0 and fingers[3] == 0 and fingers[4] ==
1:
            autopy.mouse.toggle(autopy.mouse.Button.LEFT, True)
if fingers[0] == 0 and fingers[1] == 0 and fingers[2] == 1:  # If thumb is up and index finger is down
            pyautogui.scroll(50)  # Scroll up
        elif fingers[0] == 1 and fingers[1] == 0 and fingers[2] == 0:  # If thumb is down and index finger is
up            pyautogui.scroll(-50)  # Scroll down
    else:        autopy.mouse.toggle(autopy.mouse.Button.LEFT, False)
    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)
    cv2.imshow("Image", img)
    cv2.waitKey(1)
```

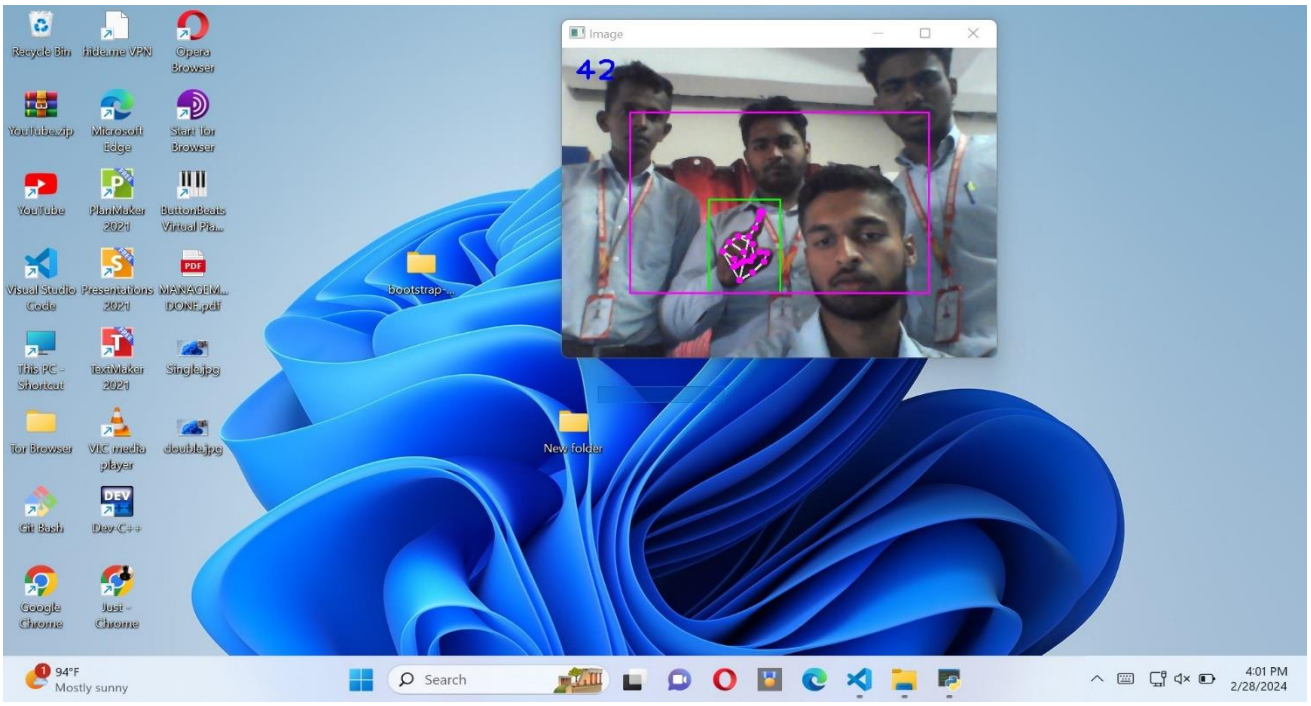
Output:



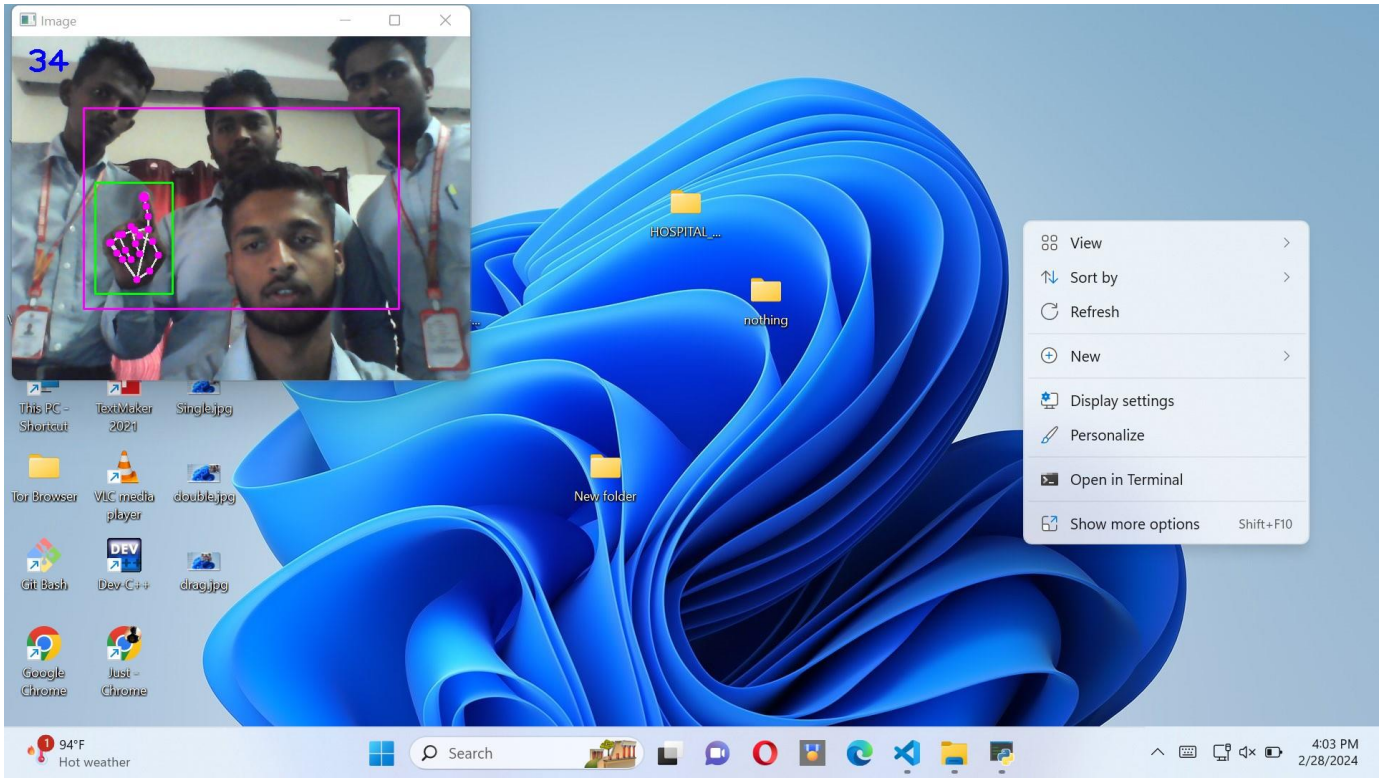
Single click



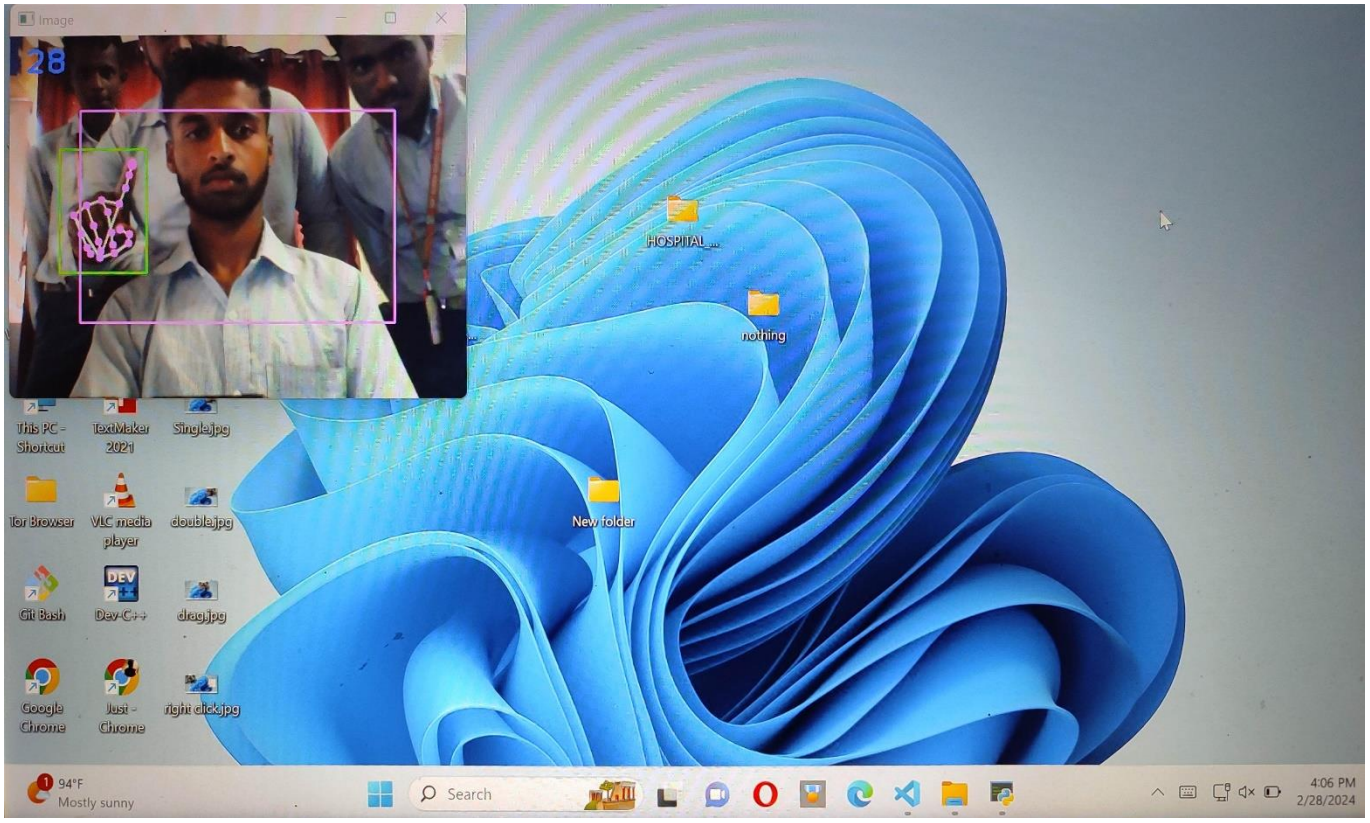
Double click



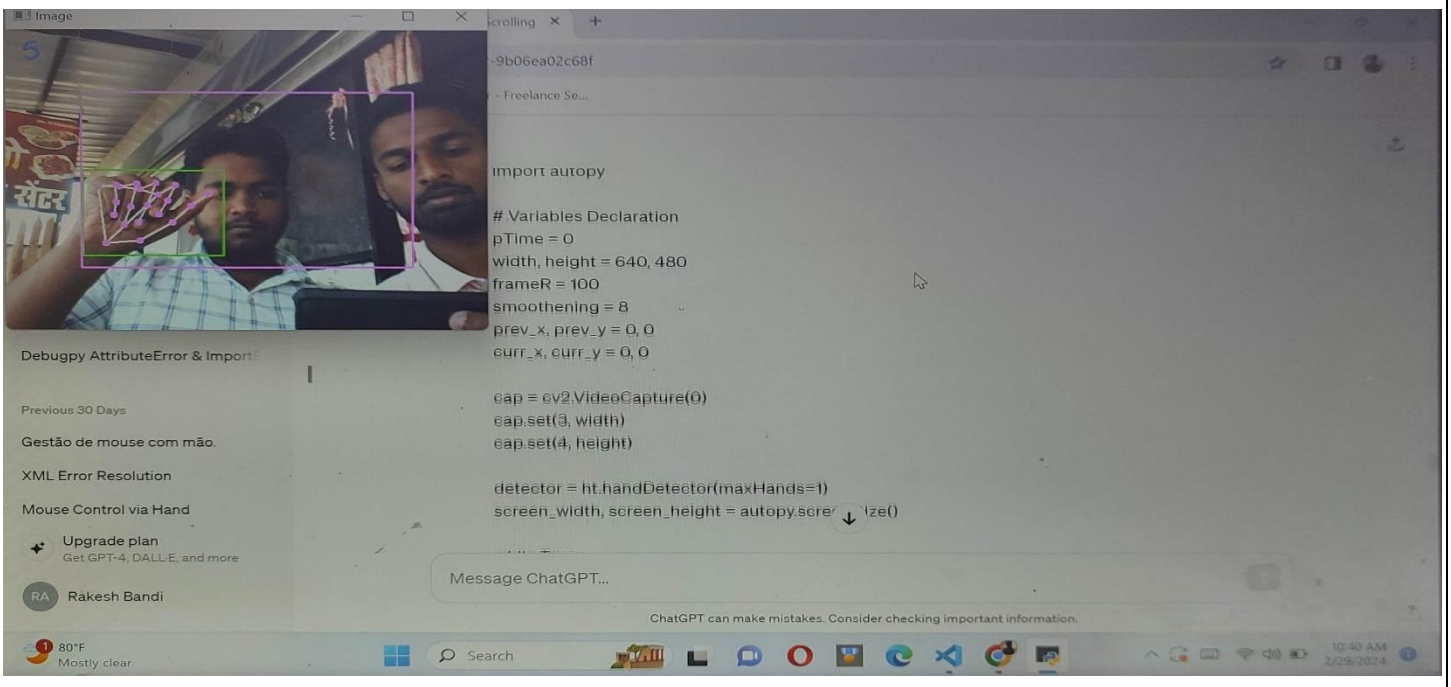
Drag and drop



Right click



Movement



Scrolling

CHAPTER 6
LIMITATION

5.1 Limitations:

- There will always be limitations of the mouse as the mouse is a hardware input device and there can be some problems like mouse click not functioning properly.
- The mouse is a hardware device like any other physical object even the mouse will have a durability time within which is functional and after its durability time we have to change the mouse.
- small decrease in accuracy in right click mouse function and some difficulties in clicking and dragging to select the text.
- It is useful only when we in the labs/room.

CHAPTER 7:
APPLICATIONS & ADVANTAGES

6.1 ADVANTAGES

1. Virtual Mouse using Hand gesture recognition allows users to control mouse with the help of hand gestures.
2. System's webcam is used for tracking hand gestures.
3. Computer vision techniques are used for gesture recognition. OpenCV consists of a package called video capture which is used to capture data from a live video.
4. Main thing we need to identify are the applications the model is going to develop so the development of the mouse movement without using the system mouse

6.2 APPLICATIONS

- Playing games

  AI virtual mice can be used to play virtual reality and augmented reality games without a physical mouse.
- Controlling the mouse

  People with hand problems can use a virtual mouse to control their computer's mouse functions.
- Reducing the spread of COVID-19

  Virtual mice can be used to reduce the spread of COVID-19 by eliminating the need for a physical mouse.

CHAPTER 8:
FUTURE SCOPE

7.1 Future Scope

The development of these techniques and models are really vast. The color detection model can be developed if we want to identify a particular color out of a colored photo. And the mouse movement can be developed in such a way it can act like a real mouse that will help us for using system without even touching the system's keyboard or mouse. The development can be in such a way it can be training on CNN's (convolutional neural networks) that will help for a better performed model. The Models can be developed in different ways by using some latest packages like 'pyautoGUI' that will help us to give commands which will identify an input and perform some function on the system. So if any separate color is detected it can perform special function or if an input from user is detected it will open any specific folder with ease without performing any actions, a simple gesture can do the job.

CHAPTER 9: CONCLUSION

8.1 Conclusion:

This model can conclude by using the topics of computer vision like open CV, it can form masks that can variate colors by using color variation techniques and also development of mouse movement by using certain packages like ‘mouse’ which will be used for the movement of mouse by using the coordinates that are linked to the detected color. This can provide ease use of systems and many other applications. So, the open CV is helping the users with different accessible forms of models that will make ease life.

` virtual mouse project implemented in Python using hand gestures represents a significant leap forward in human-computer interaction technology. By harnessing the power of computer vision and machine learning algorithms, this project offers users a novel and intuitive method of controlling their devices without the need for traditional input devices such as a physical mouse or touchpad.

CHAPTER 10: REFERENCE

9.1 Books:

1. Anna De Liddo, Ágnes Sándor, et.al, (2012). Contested Collective Intelligence: Rationale, Technologies, and a Human-Machine Annotation. Computer Supported Cooperative Work (CSCW) Volume 21, Issue 4–5, pp 417–448.
2. WAN CHUNG SON (MAY 2016). VIRTUAL MOUSE , Bachelor of Information Systems (hons) information systems engineering.

9.2 Other References:

1. http://www.w3schools.com/
2. http://stackoverflow.com/
3. https://www.tutorialspoint.com/web_development_tutorials.html
4. http://github.com/
5. http://www.vogella.com/tutorials/dotnet/artcle.html
6. https://www.btechsmartclasses.com/
7. https://chat.openai.com/