

SYNOPSIS REPORT

on

VIRTUAL MOUSE

A Project Report submitted in partial fulfilment of the requirements for the
award of the degree of

B.Tech CSE with Specialisation in CCVT

Submitted by

Atul Garg (Enroll No: R110218037)

Divyansh Thakur (Enroll No: R110218052)

Arun Tomar (Enroll No: R110218034)

Chirag Sharma (Enroll No: R110218043)

Under the guidance of

Dr. Rashmi Sharma

Assistant Professor(SG), SoCS

SCHOOL OF COMPUTER SCIENCE

UNIVERSITY OF PETROLEUM & ENERGY STUDIES

Bidholi Campus, Energy Acres, Dehradun – 248007.

January 2022

Table of Contents

Topic		Page No
Abstract		2
Table of Contents		3
1.	Introduction	4
2.	Problem Statement	5
3.	Need for the Problem	5
4.	Characteristics of Data	6
5.	Methodology	6
6.	Flow Chart	7
7.	SWOT Analysis	8
8.	Area of Focus	8
9.	Pert Chart	9
10.	References	9

ABSTRACT

This task presents a way to deal with foster a continuous hand signal acknowledgment situated in "Vision Based" that utilizes just a webcam, Python, and Computer Vision innovation, for example, picture handling that can perceive a few motions for use in PC interface collaboration.

The utilizations of continuous hand signal acknowledgment in reality are various in light of the fact that it tends to be utilized anyplace where we associate with PCs. A significant utilization of this venture is to recreate the mouse as a visual contributing gadget with each of its undertakings like left-click, right-click, double tap, relocating, and looking over.

Different applications that are applied in our task utilizing hand signal acknowledgment are making it intuitive for players to mess around by permitting them to collaborate with the game utilizing motions as opposed to utilizing a regulator, to stay away from contactless help, accordingly staying away from the spread of sicknesses.

Keywords: *Virtual Mouse, Computer Vision, Python, Hand Gestures, Voice Assistant, Contactless Support*

INTRODUCTION

Human-computer interaction is a growing field in which computer scientists study novel ways in which humans can interact with computers naturally and intuitively. One of the most widely researched topics in this field is hand gesture recognition, where hand movements are used to control computers.

Years and years prior, utilizing PCs was so difficult and complex that anyway researchers couldn't really utilize them. Yet, presently with the progression of time and to manage the require all our circles of presence, the use of PCs has wound up so natural that everyone can utilize it. As of now, a mouse or console is being utilized to associate with the PC. At times it appears to be awkward in light of the fact that individuals would rather not get off from where they are sitting or lying. A signal acknowledgment framework gives a characteristic, imaginative, and current method of non-verbal correspondence.

Motion is an indication of actual way of behaving or an outflow of sentiments. This includes the development of the body and hand. It fell into two classes: static motion and liquid signal; for the previous, body stance or hand signal indicates an image.

It has been ages since we have been utilizing hand motions for conveying in human culture. The shaking of hands, Thumbs up and Thumbs down signs have been truly existing in the climate. It is accepted that signals are the most straightforward method of collaboration with anybody. So why not matter it to the machines that we are utilizing. In this work, we are illustrating, genuine motion. The underlying arrangement incorporates a minimal expense USB web camera that can be utilized for giving the contribution to the framework. The total interaction is partitioned into 4 stages which are outline catching, picture handling, district extraction, include coordinating. To the limit, it can likewise be called as equipment since it involves a camera for following hands [5]. Point and objective of exploration work incorporate

- For most laptop touchpad is not the most comfortable and convenient.
- Main objective pre-processing is to represent the data in such a way that it can be easily interpreted and processed by the system.

- Reduce cost of hardware [2]. It focuses on extracting the features over the human hands and then matching their features to recognize the movement of the hand. Project essential feature-
- User friendly.
- Portable.
- Handle simple operation left-click dragging, minimizing.
- No hardware

The key problem in gesture interaction is how to make hand gestures understood by computers. The approaches present can be mainly divided into “Data-Glove based” and “Vision-Based” approaches. The Data-Glove-based methods use sensor devices for digitizing hand and finger motions into multi-parametric data. The extra sensors make it easy to collect hand configuration and movement. However, the devices are quite expensive and bring much cumbersome experience to the users. In contrast, the Vision-Based methods require only a camera, thus realizing a natural interaction between humans and computers without the use of any extra devices. These systems tend to complement biological vision by describing artificial vision systems that are implemented mostly in software. This approach is the cheapest, and the most lightweight. Moreover, such systems must be optimized to meet the requirements, including accuracy and robustness.

The existing system consists of a mouse that can be either wireless or wired to control the cursor, now we can use hand gestures to monitor the system. The existing virtual mouse control system consists of the simple mouse operation using the colored tips for detection which are captured by web-cam, hence colored fingers act as an object which the web-cam senses color like red, green, blue color to monitor the system, whereas could perform basic mouse operation like minimize, drag, scroll up, scroll down, left-click right-click using hand gestures without any colored finger because skin color recognition system is more flexible than the existing system. In the existing system use static hand recognition like fingertip identification, hand shape, Number of fingers to define action explicitly, which makes a system more complex to understand and difficult to use.

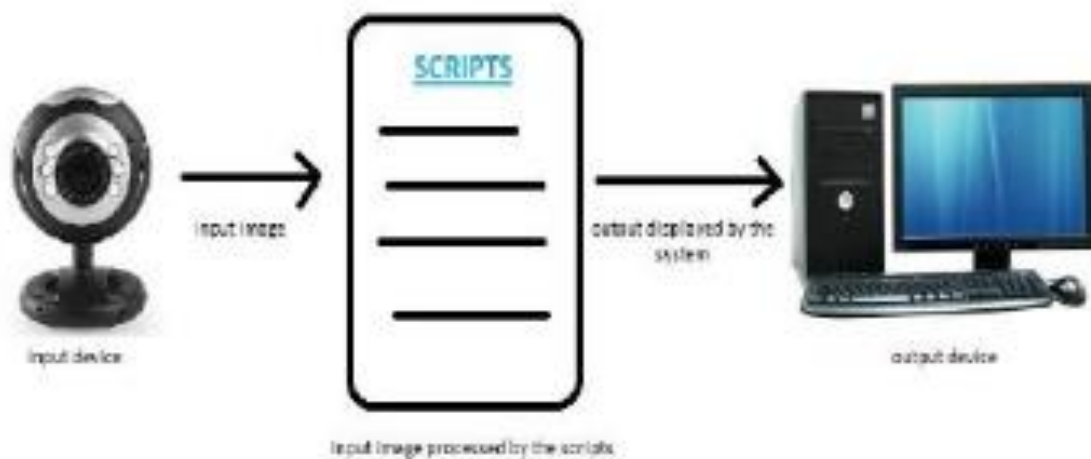


Figure. Workflow

The goal of Hand gesture recognition is to establish a complete system for detecting, recognizing, and interpreting gesture recognition through computer vision using Python and OpenCV techniques. To establish a people-friendly system to aid in the ease of living of the challenged people. Then, the palm and fingers are segmented to detect and recognize the fingers. Finally, a rule classifier is applied to predict the labels of hand gestures.

The scope of this project is to create a method to recognize hand gestures, based on a pattern recognition technique. Hand gestures can be used as an alternative input device that enables interaction with a computer without a mouse or keyboard, such as dragging, dropping, and moving files through the desktop environment, as well as cut and paste operations. Moreover, they can be used to control slide show presentations.

PROBLEM STATEMENT

The objective of Virtual Mouse is to lay out a total framework for identifying, perceiving, and deciphering signal acknowledgment through PC vision utilizing Python and OpenCV methods. To lay out a human cordial framework to help with the simplicity of living of the tested individuals, and furthermore through this task, there will be a decrease in the spread of illnesses and infections.

- A) To make a virtual mouse to take care of the issue of:
- B) The spread of infections among clients, similar to the COVID-19 spread, i.e, offering contactless help to the client.
- C) That is looked by the tested individuals of utilizing an actual mouse.

The proposed AI virtual mouse framework can be utilized to conquer issues in reality, for example, circumstances where there is no space to utilize an actual mouse and furthermore for the people who have issues in their grasp and can't handle an actual mouse. Likewise, in the midst of the COVID-19 circumstance, it isn't protected to utilize the gadgets by contacting them since it might bring about a potential circumstance of spread of the infection by contacting the gadgets, so the proposed AI virtual mouse can be utilized to defeat these issues since hand motion and hand Tip identification is utilized to control the PC mouse capacities by utilizing a webcam or an underlying camera.

Objective

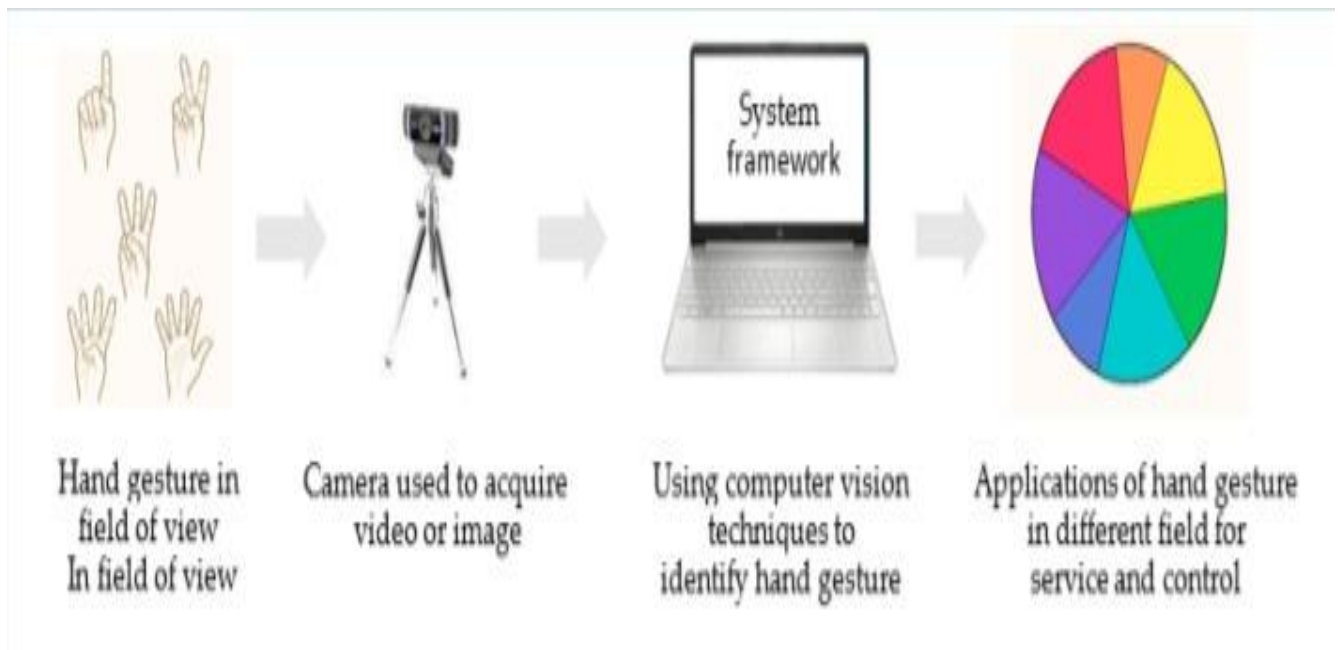
- The principal objective of the proposed AI virtual mouse framework is to foster an option in contrast to the ordinary and conventional mouse framework to perform and control the mouse capacities, and this can be accomplished with the assistance of a web camera that catches the hand motions and hand tip and afterward processes these edges to play out the specific mouse capacity like left click, right snap, and looking over work.
- We want to make a framework that can recognize explicit human motions and use them to pass on data for gadget control and by carrying out ongoing signals acknowledgment a client have some control over a PC by doing a particular signal before a camcorder connected to a PC. Here the framework can be worked utilizing hand motions without utilizing a console and mouse.
- To plan a virtual mouse that recognizes hand motion designs rather than an actual mouse. To lay out a total framework for distinguishing, perceiving and deciphering motion acknowledgment utilizing Python and OpenCV procedures.

NEED FOR THE PROBLEM

In the present digitized world, handling speeds have expanded emphatically, with PCs being progressed to the levels where they can help people in complex errands. However, input innovations appear to cause a significant bottleneck in playing out a portion of the undertakings, under-using the accessible assets and confining the expressiveness of use. Hand Gesture acknowledgment acts the hero here. PC Vision strategies for hand motion interfaces should outperform current execution as far as vigor and speed to accomplish intuitiveness and convenience.

As we probably are aware, in the present time, there is a spread of an infection COVID-19, and its contact is prompting the introduction of new variations, utilizing frameworks that work without making any contact is significant.

CHARACTERISTICS OF DATA



- **INPUT:** Single or Double Hands
- **OUTPUT:** Specific function is performed, like a double click, dragging, volume control, brightness control, etc.

METHODOLOGY

Our task utilizes Otsu Thresholding Algorithm. This calculation regards any division issue as a grouping issue. The complete picture level is isolated into two classes one is hand and the other is foundation. The limit not entirely settled by figuring the proportion between class difference and absolute class change. A separating technique is utilized to actually eliminate foundation and item commotion in the fragmented picture. This strategy comprises of opening, and shutting tasks.

The Camera Used in the AI Virtual Mouse System. The proposed AI virtual mouse framework depends on the edges that have been caught by the webcam in a PC or PC. By utilizing the Python PC vision library OpenCV, the video catch object is made and the web camera will begin catching video, as displayed in Figure 4. The web camera catches and passes the casings to the AI virtual framework.

Catching the Video and Processing. The AI virtual mouse framework utilizes the webcam where each edge is caught till the end of the program. The video outlines are handled from BGR to RGB variety space to find the hands in the video outline by outline.

Rectangular Region for Moving through the Window. The AI virtual mouse framework utilizes the groundbreaking calculation, and it changes over the directions of fingertip from the webcam screen to the PC window full screen for controlling the mouse. Whenever the hands are identified and when we find which finger is ready for playing out the specific mouse work, a rectangular box is attracted regarding the PC window in the webcam area where we move all through the window utilizing the mouse cursor.

Recognizing Which Finger Is Up and Performing the Particular Mouse Function. In this stage, we are recognizing which finger is up utilizing the tip Id of the individual finger that we observed utilizing the MediaPipe and the separate co-ordinates of the fingers that are up, as displayed in Figure 6, and as per that, the specific mouse work is performed.

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

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 PC utilizing the AutoPy bundle of Python.

For the Mouse to Perform Left Button Click. If 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.

For the Mouse to Perform Right Button 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

Mouse cursor moving around the computer window. lesser than 40 px, the computer is made to perform the right mouse button click using the pynput Python package.

The distance between the two fingers is more noteworthy than 40 px and assuming the two fingers are climbed the page, the PC is made to play out the scroll up mouse work utilizing the PyAutoGUI Python bundle.

For the Mouse to Perform Scroll down Function. If both the file finger with tip Id 1 and the center finger with tip Id 2 are up and the distance between the two fingers is more prominent than 40px and on the off chance that the two fingers are dropped down the page, the PC is made to play out the look down mouse work utilizing the PyAutoGUI Python bundle.

For No Action to be Performed on the Screen. On the off chance that all the fingers are up with tip Id 0, 1, 2, 3, and 4, the PC is made to not play out any mouse occasions in the screen.

For the Mouse to Perform Scroll up Function. If both the list finger with tip Id 1 and the center finger with tip

The client performs explicit signals by single or two hands before the camera which interface with the framework structure that includes various potential procedures to separate elements and characterize hand motions to have the option to control a few potential applications.

The venture will follow steps:

- Picking the signals for playing out the assignments, wherein, the hand motions will attempt to perform errands like moving the cursor, double tap, drag, and drop, and so forth.
- It will execute these motions utilizing the calculation and open PC vision methods.

OPENCV

It's a library of Python ties intended to tackle PC vision issues. It is utilized for picture handling and performing PC vision undertakings. It upholds a wide assortment of programming dialects like Python, C++, Java, and so on. It assists us with gaining picture handling from essentials to progress. Assuming that it gets incorporated with numpy, other development libraries and profoundly streamlined for mathematical tasks, then, at that point, the quantity of weapons increments.

PyAutoGUI

It is a cross-stage GUI robotization Python module for people. Used to automatically control the mouse and console.

The three significant working frameworks each have various approaches to control the mouse and console automatically. This can frequently include befuddling, and profoundly specialized subtleties. The occupation of PyAutoGUI is to conceal all of this intricacy behind a straightforward API.

- On macOS, PyAutoGUI utilizes the rubicon-objc module to get to the Cocoa API.
- On Linux, PyAutoGUI utilizes the Xlib module to get to the X11 or X Window System.
- On Windows, PyAutoGUI gets to the Windows API (additionally called the WinAPI or win32 API) through the underlying ctypes module. The nicewin module at <https://github.com/asweigart/nicewin> gives an exhibition to how Windows API calls can be made through Python.

Pynput

A library allows us to control and monitor/listen to your input devices such as they keyboard and mouse. Allows us to monitor the mouse. A library of Python that can be used to capture keyboard inputs.

It contains subpackages for each type of input device supported:

`pynput.mouse`

Contains classes for controlling and monitoring a mouse or trackpad.

`pynput.keyboard`

Contains classes for controlling and monitoring the keyboard.

A mouse listener is a `threading.Thread`, and all callbacks will be invoked from the thread.

The listener callbacks are invoked directly from an operating thread on some platforms, notably Windows.

If a callback handler raises an exception, the listener will be stopped.

This means that long running procedures and blocking operations should not be invoked from the callback, as these risks freezing input for all processes.

A possible workaround is to just dispatch incoming messages to a queue, and let a separate thread handle them.

If a callback handler raises an exception, the listener will be stopped. Since callbacks run in a dedicated thread, the exceptions will not automatically be reraised.

The segmentation of gray scale image of a hand gesture is performed using Otsu thresholding algorithm.

Otsu Thresholding Algorithm

- It regards any division issue as characterization issue
- All out-picture level is isolated into two classes one is hand and other is foundation.
- Edge not entirely settled by registering the proportion between class difference and all out-class fluctuation.
- A sifting strategy is utilized to actually eliminate foundation and item commotion in the portioned picture.
- This technique comprises of opening, and shutting activities.

Design

Step 1: Choose Gestures for performing errands

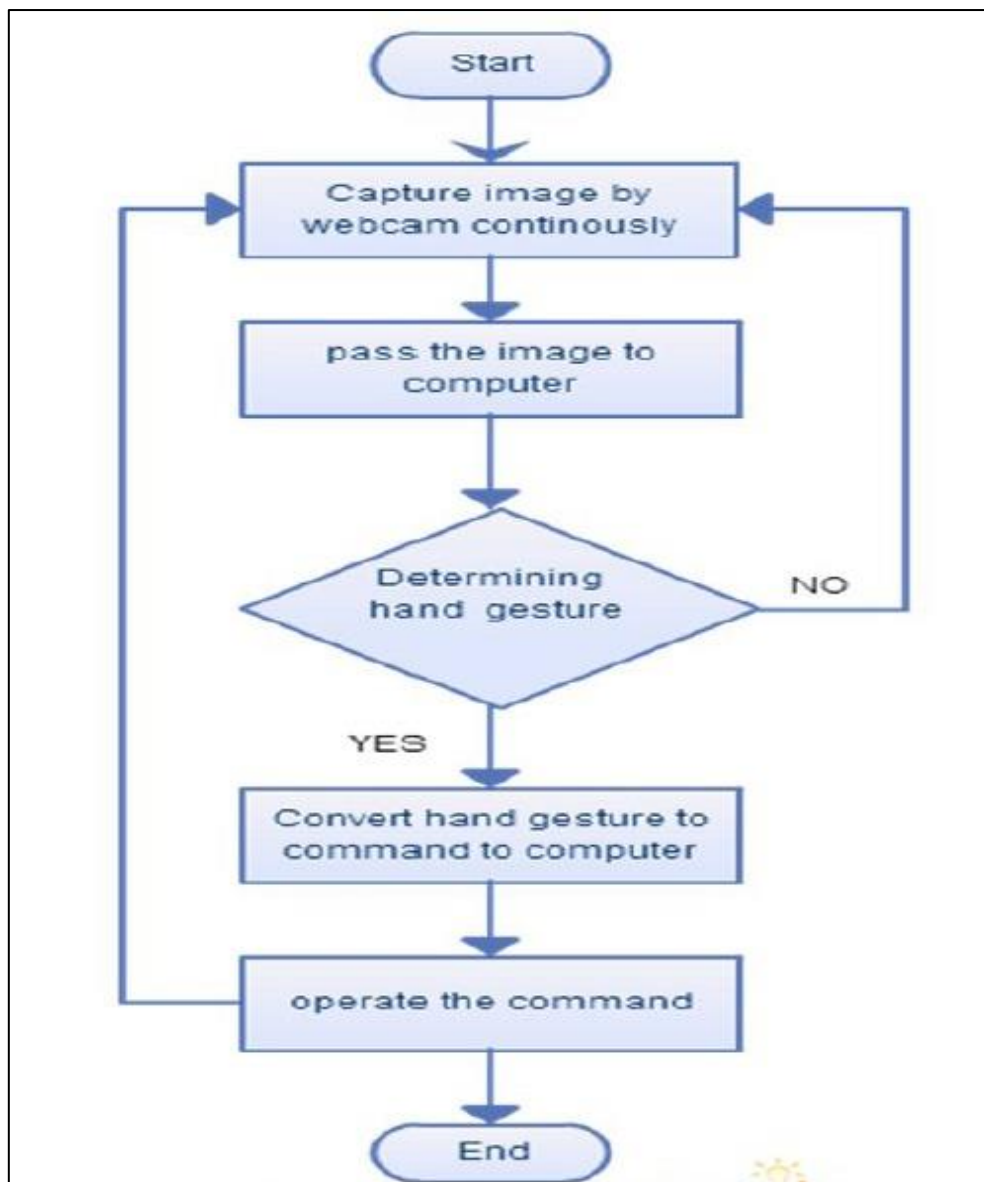
- The initial phase in the task was to pick signals that would be utilized for the control.
- The signals ought to be straightforward, straightforward, and simple for normal individuals to utilize. Likewise, the motions ought to be different enough from one another so planning a few one-of-a-kind elements for every one of them for motivations behind object recognition would be simple

Step 2: Implement the Algorithms Using OpenCV

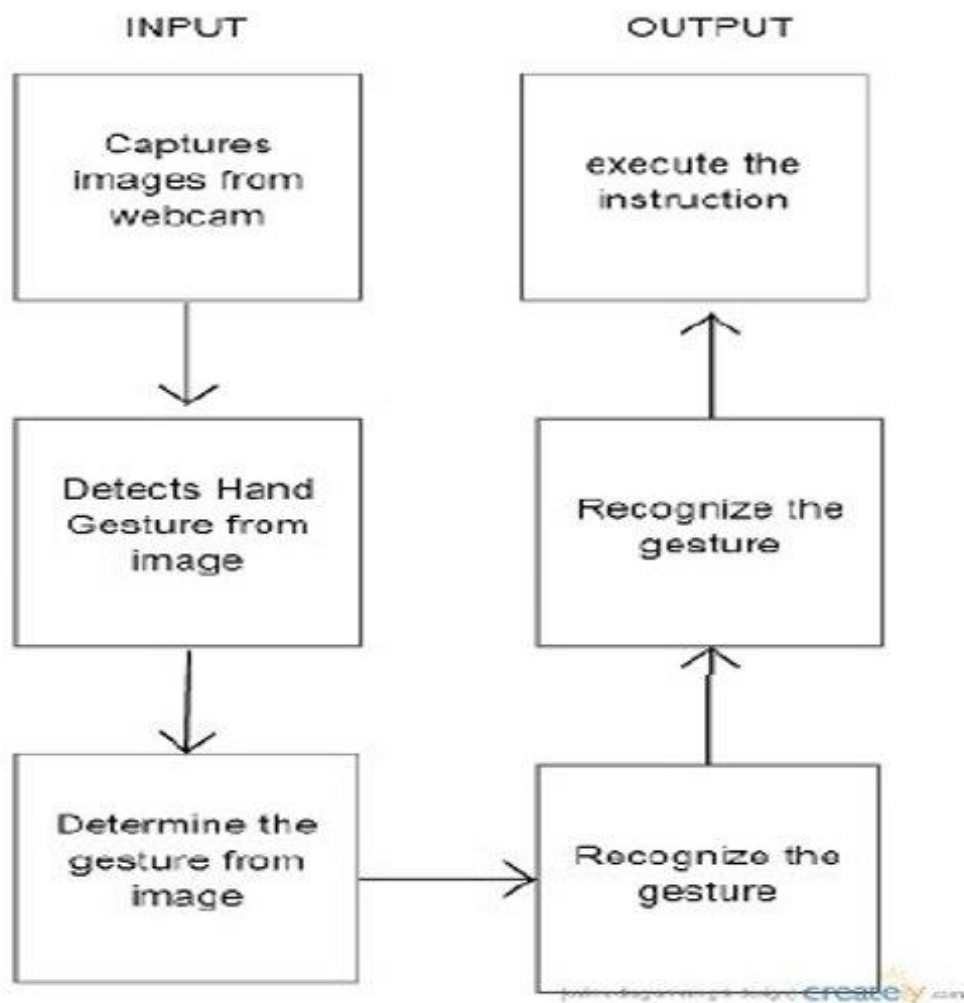
The subsequent stage was to execute the required calculations to handle the pictures and to perceive the signals, utilizing OpenCV. The subsequent program ought to do the accompanying system:

- catch the signals through a web camera
- produce a picture in anticipation of marking and hand tourist spots recognition
- remove highlights from the Media Pipe for signal acknowledgment
- yield the perceived signal/order

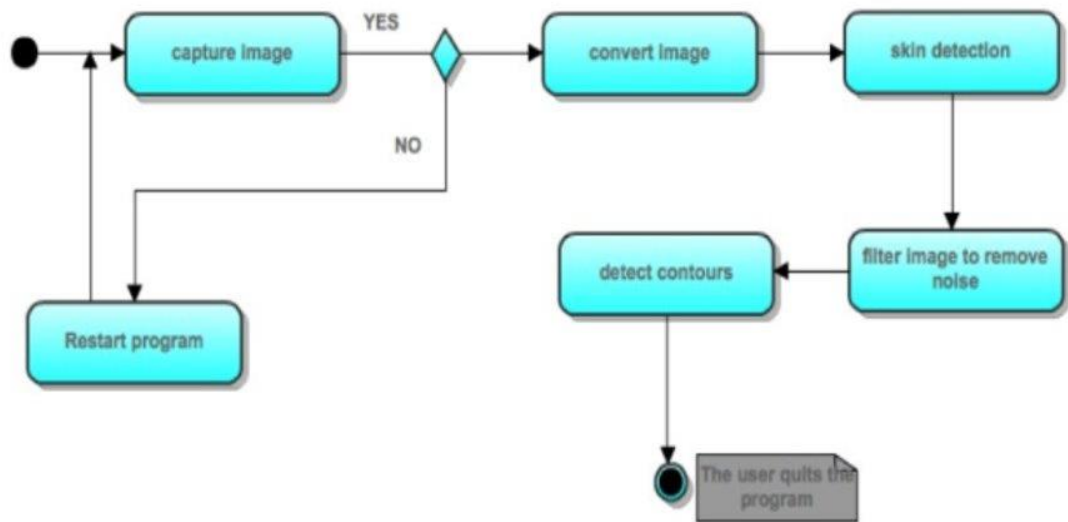
FLOWCHART



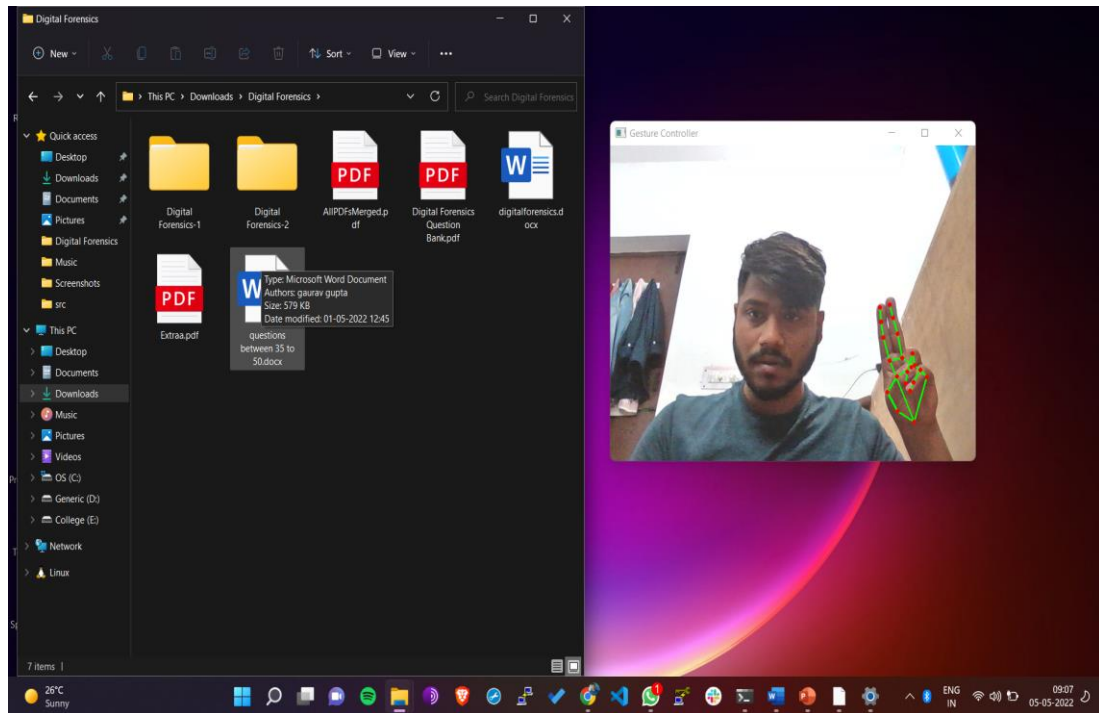
Block Diagram



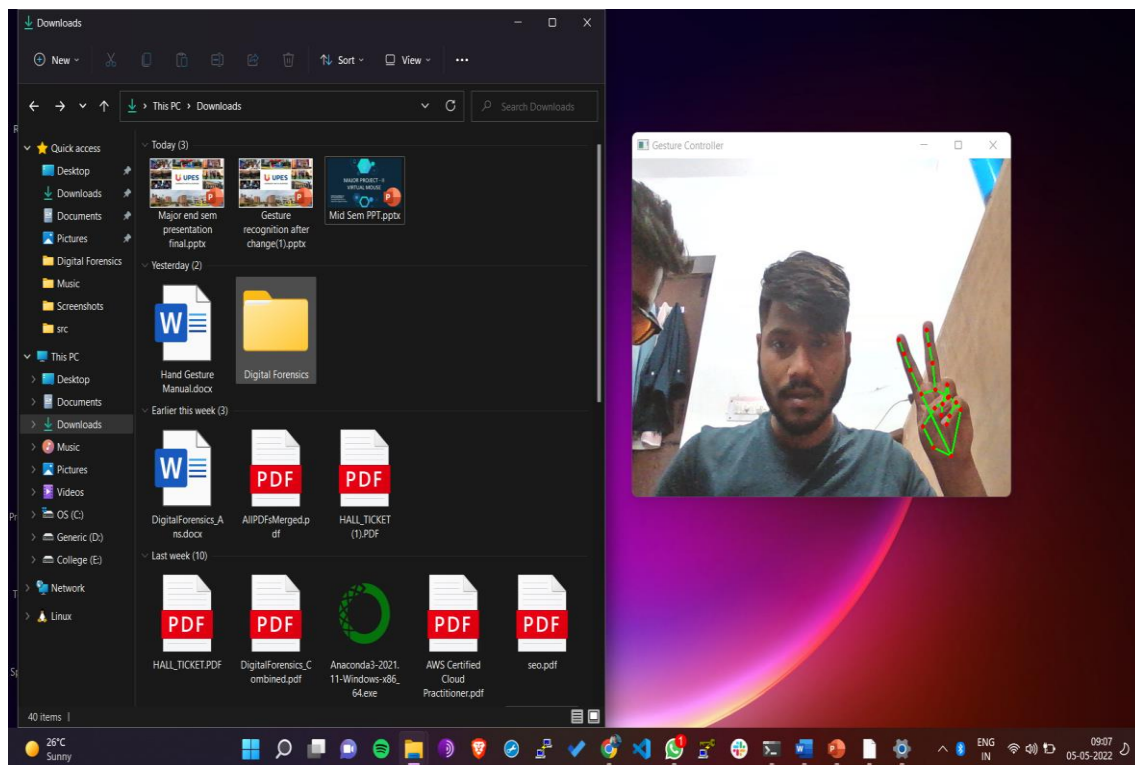
Activity Diagram



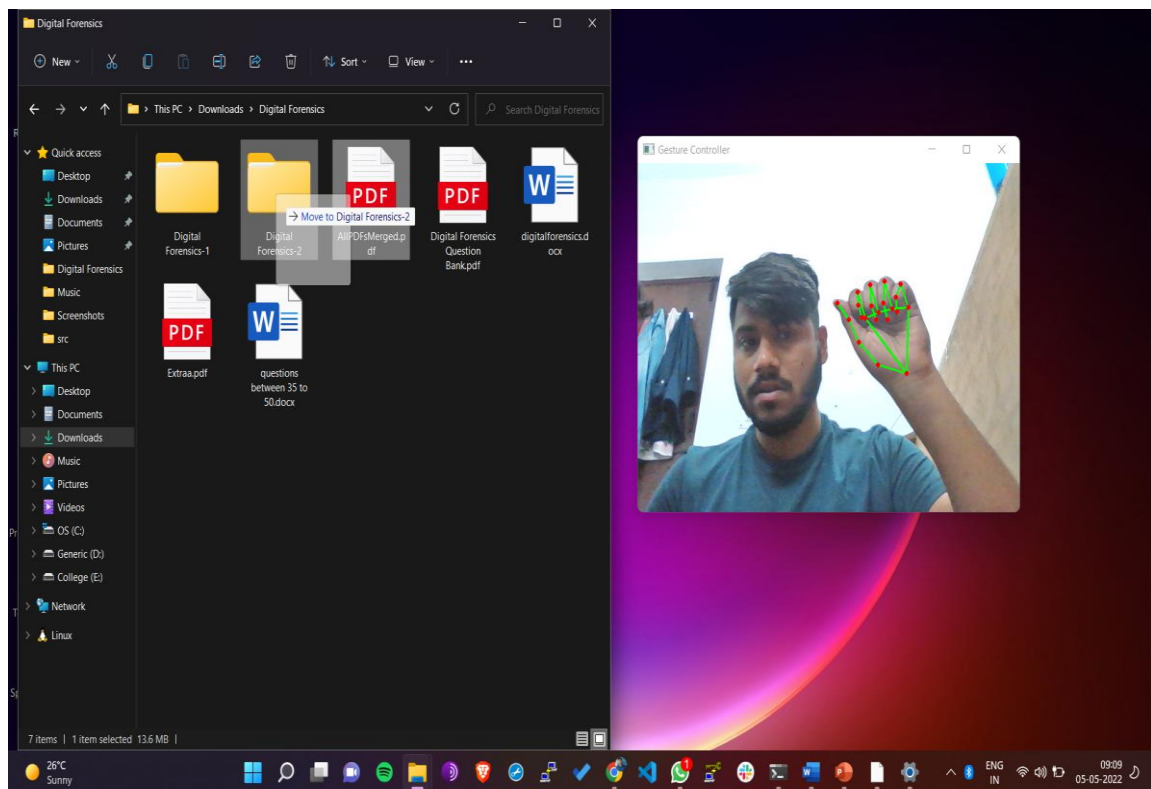
OUTPUT



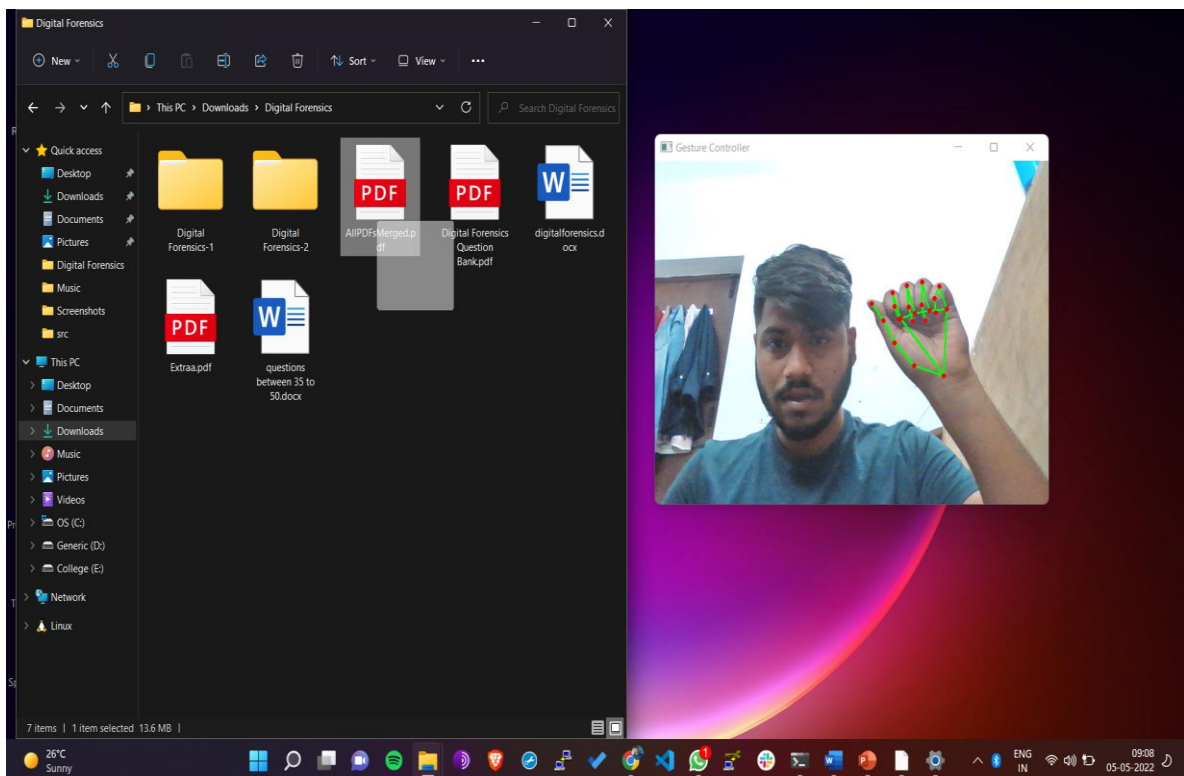
Double Click

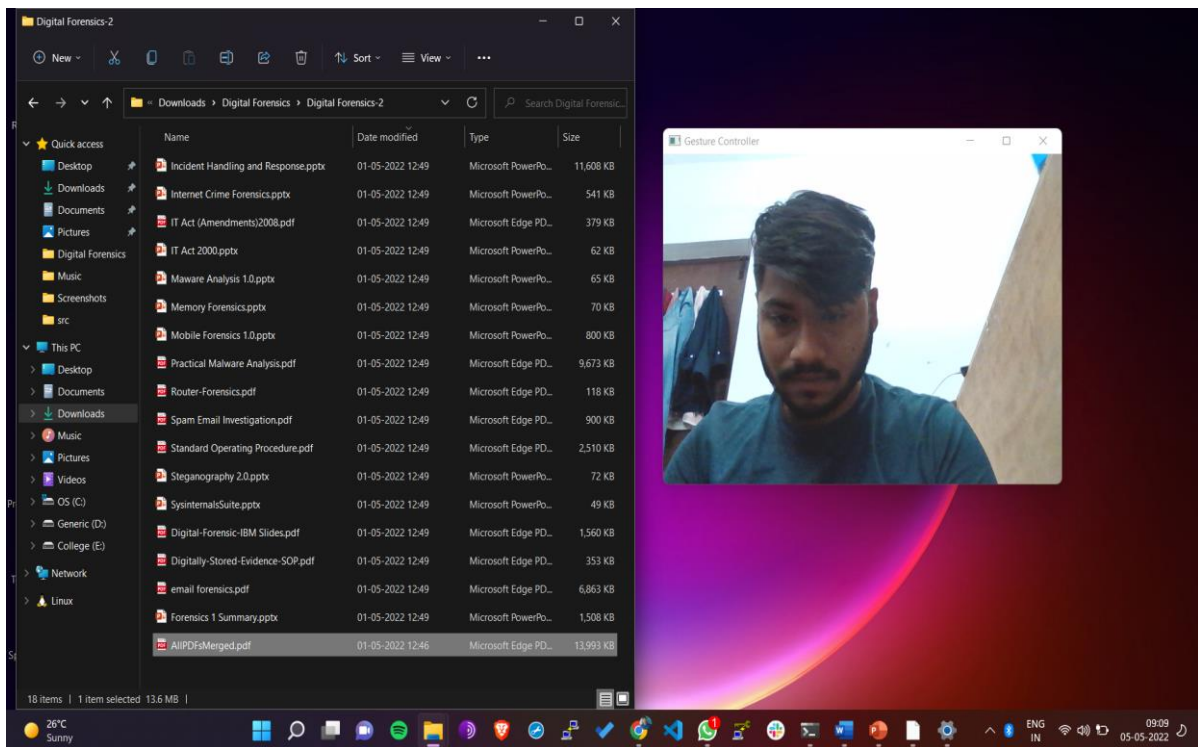


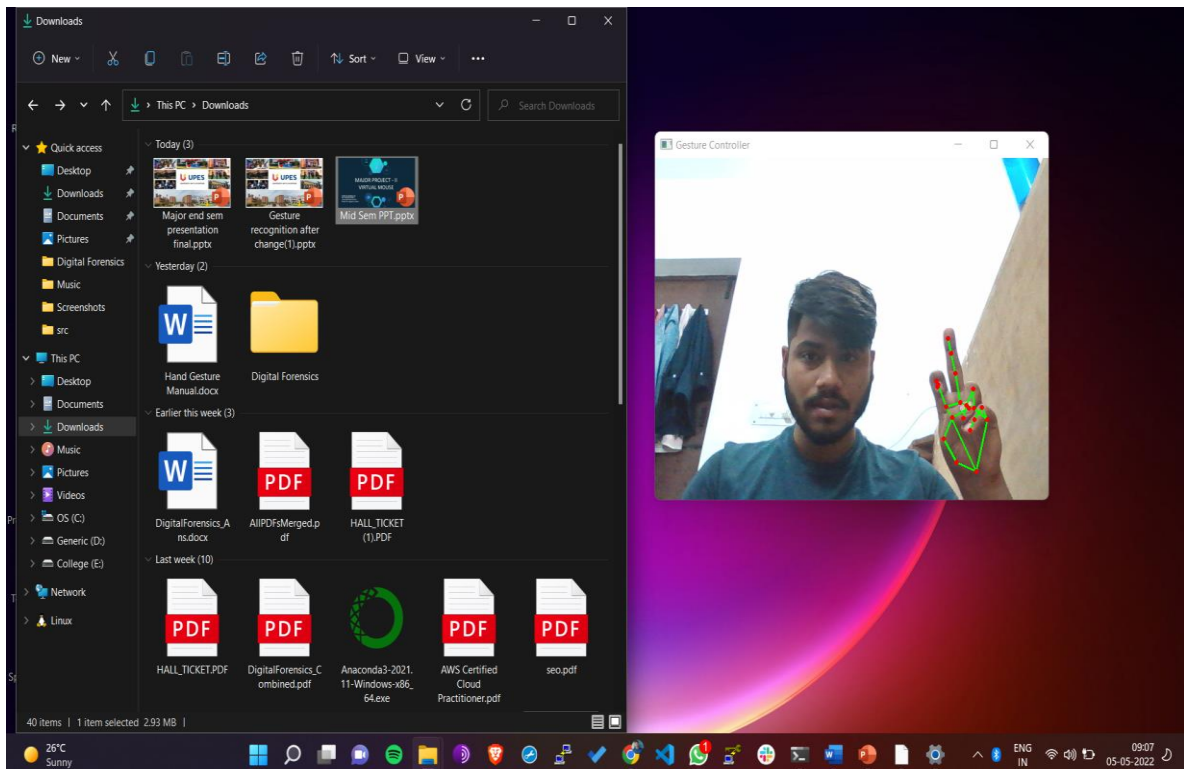
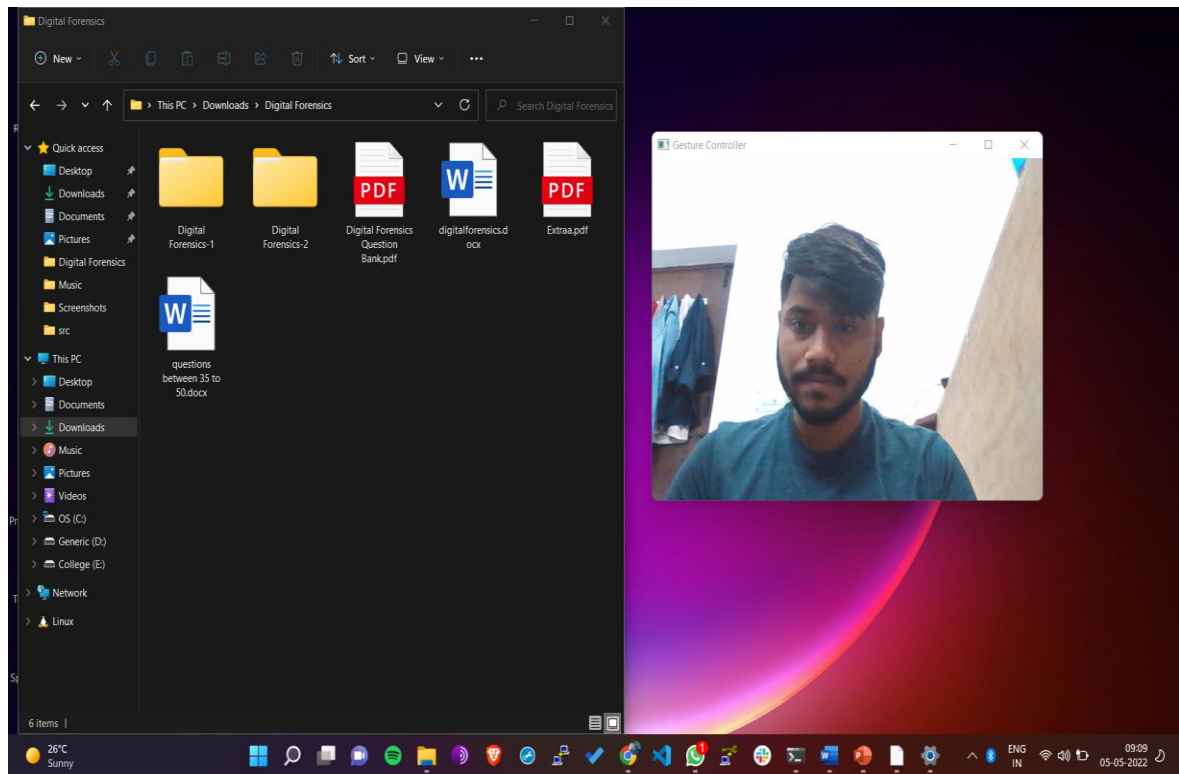
Movement



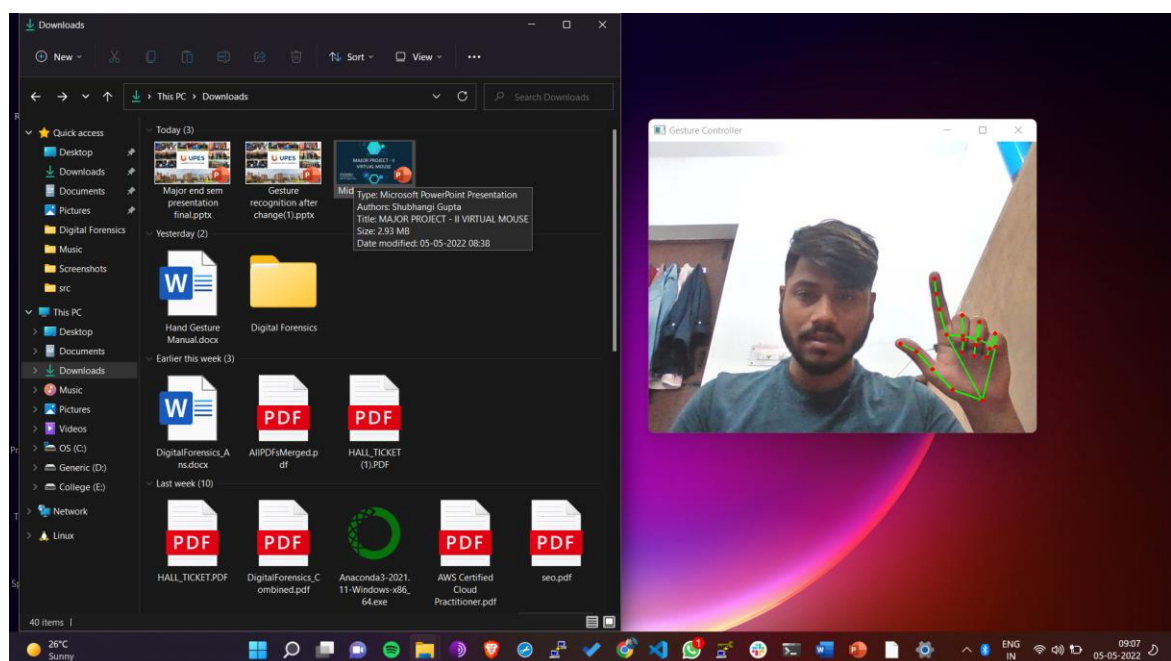
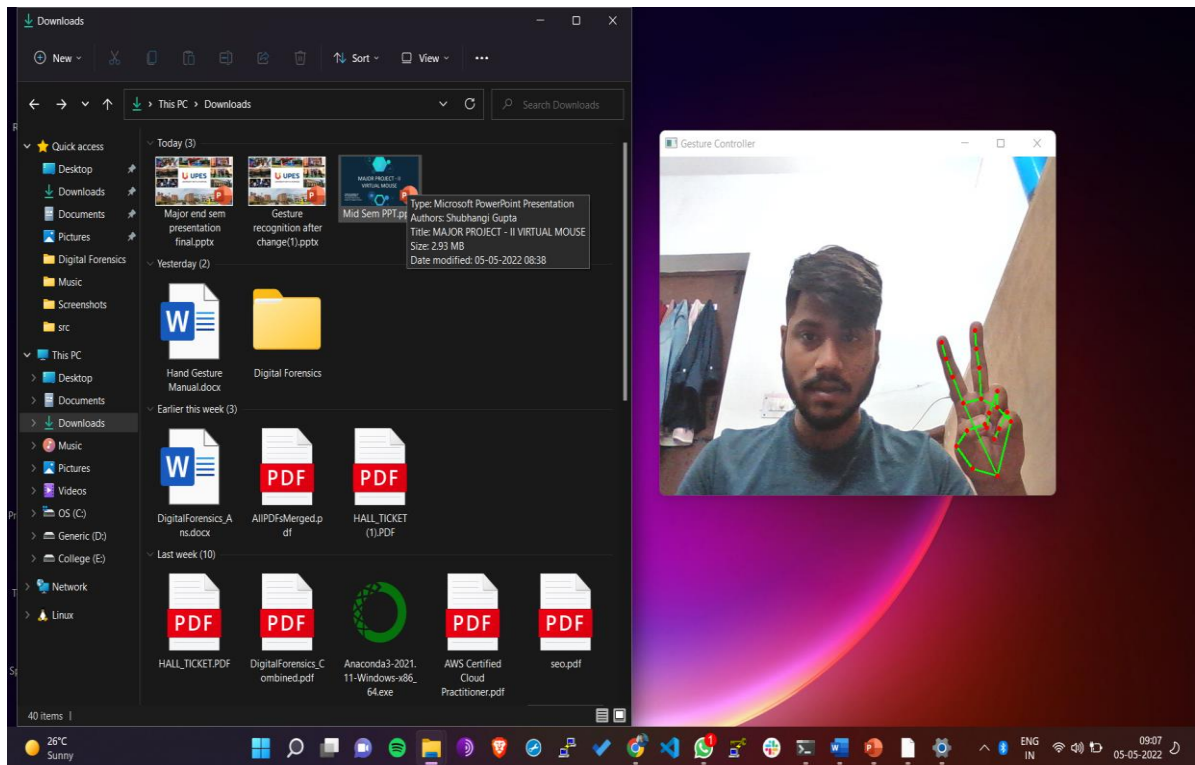
Drag



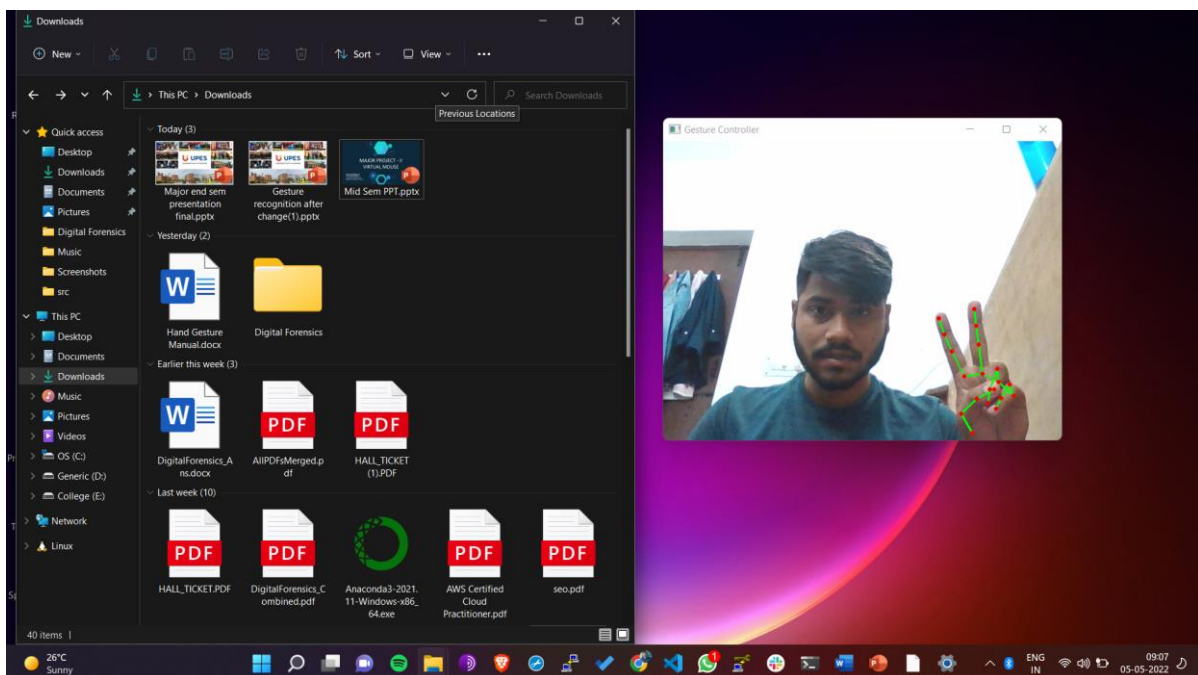
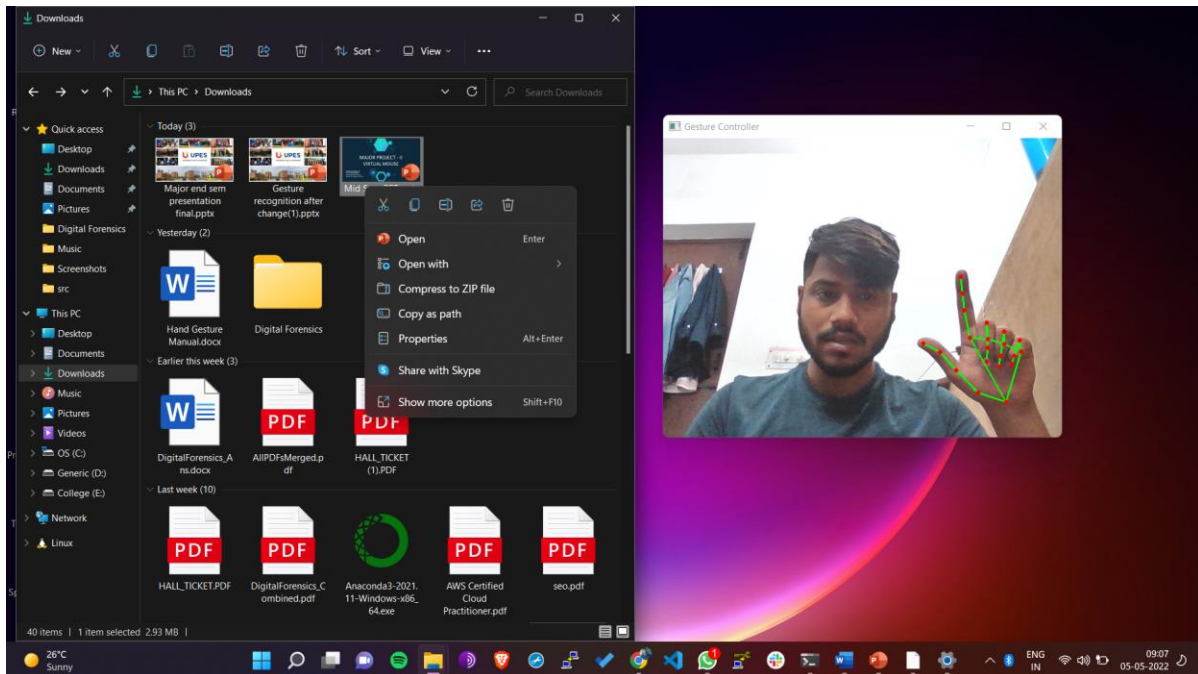




Left Click

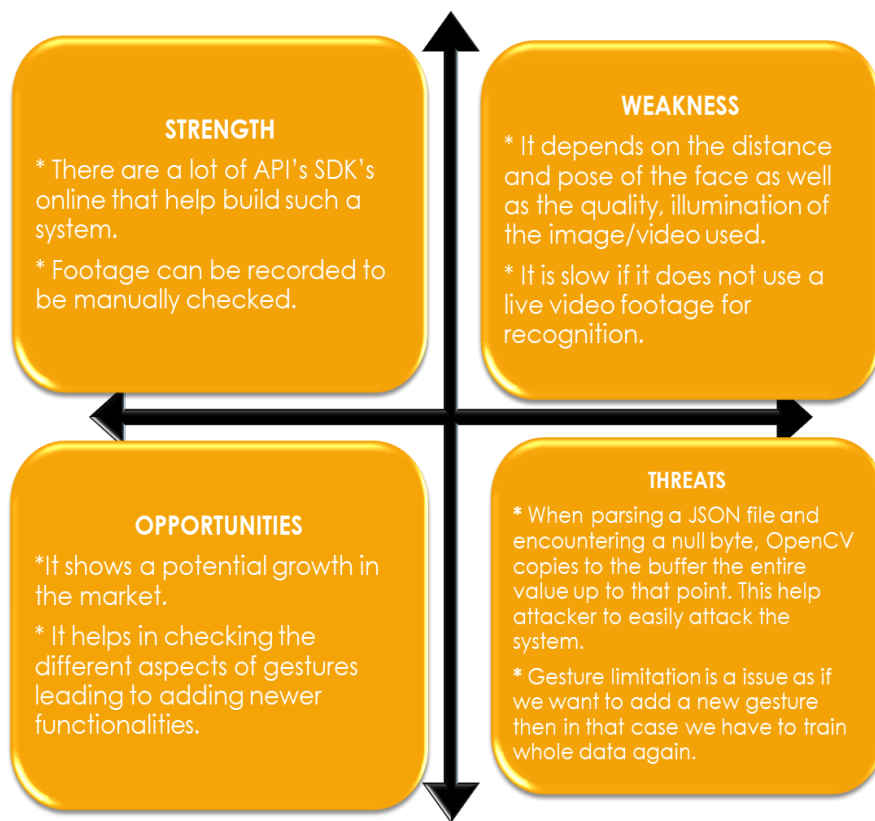


Right Click



SWOT ANALYSIS

Swot is referred as Strengths, Weaknesses, Opportunities, and Threats. A method used to identify and analyze internal strengths and weaknesses and external.

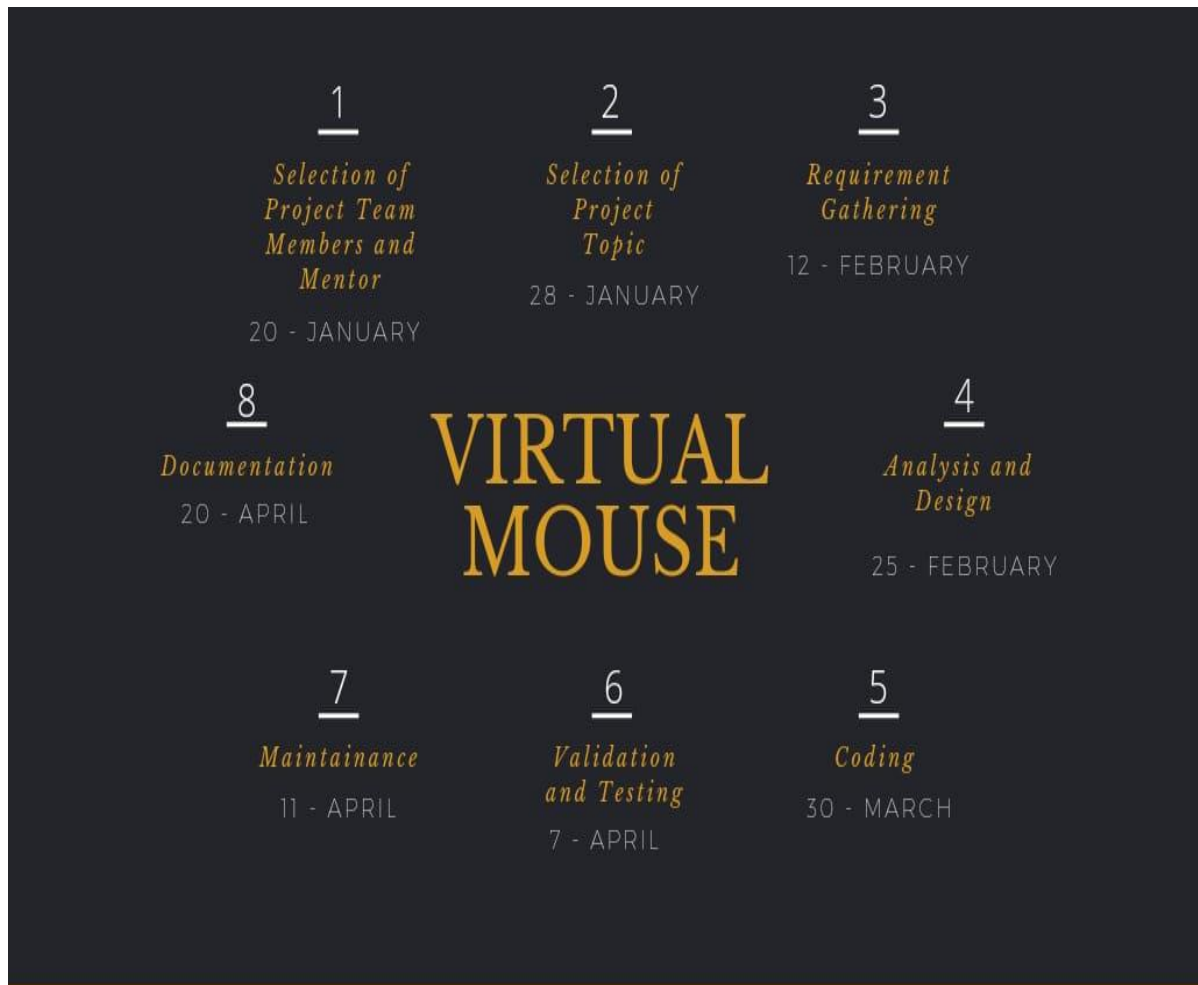


AREA OF FOCUS

This task has different applications like:

- **Contactless Support:** It works without making any actual contacts, so stays away from the spreading of infections among clients, similar to the pandemic-COVID-19.
- **Communication through signing:** It's utilized in communication through signing as a straightforward motion with one hand having similar significance all around the world and means either 'greetings' or 'farewell'.
- **Intuitive Platform:** Provides an intelligent stage for the clients to mess around, convey, talk, and so forth.
- Clinical and Health
- Automated control
- Gaming

PERT CHART



REFERENCES

- <http://www.cs.iit.edu/~agam/cs512/lect-notes/opencv-intro/opencv-intro.html>
- <http://ccdware.com/Files/CCDStack%20Tutorial.pdf>
- http://www710.univ-lyon1.fr/~bouakaz/OpenCV-0.9.5/docs/ref/OpenCVRef_ImageProcessing.htm
- <http://opencv.willowgarage.com/wiki/>
- <https://aihubprojects.com/hand-detection-gesture-recognition-opencv-python/#:~:text=Hand%20gesture%20recognition%20is%20one,laptops%2C%20smart%20phones%2C%20etc.>
- https://www.google.com/search?q=uses+of+gesture+recognition+in+python&rlz=1C1CHBD_enIN888IN888&ei=Em9yYu_vGdaf4-EPja2fwAc&oq=uses+of+gesture+recognition+in+&gs_lcp=Cgdnd3Mtd2l6EAMYA DIICCEQFhAdEB4yCAghEBYQHRAeMggIIRAWEB0QHjIICCEQFhAdEB4yCAghEBYQHRAeMggIIRAWEB0QHjIICCEQFhAdEB4yCAghEBYQHRAeMggIIRAWEB0QHjIICCEQFhAdEB46BwgAEecQsAM6BggAEBYQHkoFCDwSATFKBAhBGABKBAhGGABQmgJYnwhgiRNoAXABeACAAdgBiAH5BZIBBTauMy4xmAEAoAEByAEIwAEB&sclient=gws-wiz
- https://www.google.com/search?q=swot+analysis&rlz=1C1CHBD_enIN888IN888&oq=SWOT+an&aqs=chrome.0.0i433i512j69i57j0i433i512j0i512j0i433i512l2j0i512l2j0i433i512j0i512.2873j0j7&sourceid=chrome&ie=UTF-8
- <https://pynput.readthedocs.io/en/latest/>
- <https://pypi.org/project/pynput/>
- <https://www.geeksforgeeks.org/how-to-use-pynput-to-make-a-keylogger/>

- <https://pypi.org/project/PyAutoGUI/>
- <https://learnopencv.com/getting-started-with-opencv/>
- <https://www.geeksforgeeks.org/introduction-to-opencv/>
- <https://pythonprogramming.net/loading-images-python-opencv-tutorial/>
- <https://www.geeksforgeeks.org/opencv-python-tutorial/>