Hand Gesture Controller (Virtual Mouse) and Voice Assistant using OpenCV, ML, Python

Gangurde Shipranjali Kiran[1], Pallavi Somnath Avhad[2], RautShubham Shantaram[3],

Sonawane Snehal Shashikant[4], Prof. Pratibha V.Waje[5]

Information Technology, Sir Visvesvaraya Institute of Technology, Nashik[1-4]

Guide, Information Technology, Sir Visvesvaraya Institute of Technology, Nashik[5]

Abstract: The mouse is one of the useful hardware inventions of Human-Computer Interaction (HCI) technology. Currently, Bluetooth mouse or a wireless mouse these devices are used but the main disadvantage of these devices is it consumes a battery for power and it requires dongle/ adaptor to connect it to the PC. In the proposed Hand Gesture Controller (Virtual Mouse) and Voice Assistant using OpenCV, ML, Python. This limitations will be overcome by introducing webcam or a built-in camera for capturing of hand gestures and hand tip detection using computer vision. The algorithm used in the system makes use of the machine learning algorithm is Based on the hand gestures, the computer can be controlled virtually and can perform right click, left click, scroll-up, scroll-down and computer cursor function without any physical interaction. The algorithm is based on deep learning for detecting the hand gesture. Hence, the proposed system will avoid full physical interaction with voice assistant for controlling all computer functions. which is helpful for COVID-19 spread by eliminating the human intervention and dependency of devices to control the computer.

Keywords: Hand Gesture, Virtual, OpenCV, Machine Learning, MediaPipe, mouse cursor

I.   INTRODUCTION

With the development of different technologies in the areas of virtual devices which we uses in our daily life, these devices becomes compact or invisible in the form of wireless or Bluetooth technologies. This paper proposes an Hand Gesture Controller (Virtual Mouse)that makes use of the hand gestures and hand tip detection for performing mouse functions in the computer system and voice assistant for controlling computer functions using computer vision or python. The main objective of the proposed system is to perform mouse cursor operations and scrolling function using a web camera or a built-in camera in the computer instead of using a traditional wired or wire-less mouse device. Hand gesture and hand tip detection by using computer vision is used as a Human Computer Interaction with the computer. With the use of the Hand Gesture Controller (Virtual Mouse), we can track the finger tips of the hand gesture by using a built-in camera or web camera and perform the mouse cursor operations and scrolling function and also moving the cursor around the window with it.

While using a Bluetooth or a wireless mouse, some devices such as the mouse, the adaptor connect to the PC, and also, it consumes battery to power the mouse to operate, but in this paper, the user uses his webcam and uses his hand gestures to control the computer mouse operations. In the proposed system, the web camera captures the gesture and movement of figers 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 language which is most popular programming language is used for developing the Hand Gesture Controller (Virtual Mouse) and voice assistant, and also, OpenCV which is the library of computer vision is used in the Hand Gesture Controller (Virtual Mouse). In the proposed Hand Gesture Controller (Virtual Mouse), the model use the MediaPipe framework for the tracking of the hands and for tracking of the hand tip. It also uses packeges such as Pynput, Autopy, and PyAutoGUI for moving around the window screen of the computer for performing functions such as right click, left click, and scrolling functions. The results of the proposed model gives high accuracy, and the proposed model will works with flexiblity in real-world application on computer system without the use of a GPU.

II.   LITERATURE SURVEY

There are some related works carried out on virtual mouse using hand gesture detection by wearing a glove in the hand and also using color tips in the hands for gesture recognition, but they are no more accurate in mouse functions. The recognition is not so accurate because of wearing gloves; also, the gloves are also not suited for some users, and in some cases, the recognition is not so accurate because of the failure of detection of color tips. Some efforts have been made for camera-based detection of the hand gesture interface.

Sr.no	Year	Proposed Model	Limitations
1.	1990	Quam introduced an early hardware-based system; in the system, the user should wear a DataGlove.	Difficult to perform some of the gesture controls using the system.
2.	2010	A Real-Time Hand Gesture Recognition System Using Motion History Image.	More complicated hand gestures
3.	2015	Cursor Control System Using Hand Gesture Recognition	Stored frames are needed to be processed for hand segmentation and skin pixel detection.
4.	2018	Cursor Control using Hand Gestures. The system proposes the different bands to perform different functions of the mouse.	Depends on various colors to perform mouse functions.
5.	2021	Virtual Mouse Using Hand Gesture" where the model detection is based on colors.	Only few mouse functions are performed.

III.   SYSTEM OVERVIEW

1.MediaPipe:

MediaPipe is an opensource framework of Google. MediaPipe is a framework is used for applying in a machine learning pipeline. The MediaPipe framework is useful cross platform for development since the framework is built using the time series data. The MediaPipe framework is multimodal, this framework can be applied to various videos and audios.

The MediaPipe framework will be used by the developer for analyzing and building the system through graphs, and it also used for developing the systems for the application purpose. The steps involved in the system that uses MediaPipe are carried out in the pipeline configuration. The pipeline created can run in various platforms allowing scalability in mobile and desktops.

The MediaPipe framework is based on 3 fundamental parts; they are performance evaluation, framework for retrieving sensor data, and a collection of components which are called calculators, and they are reusable. A pipeline is a graph consists of components called calculators, where each calculator is connected by streams in which the packets of data flow through. Developers are able to replace or define custom calculators anywhere in the graph creating their own application. The calculators and streams combined create a data-flow diagram; the graph is created with MediaPipe where each node is a calculator and the nodes are connected by streams.
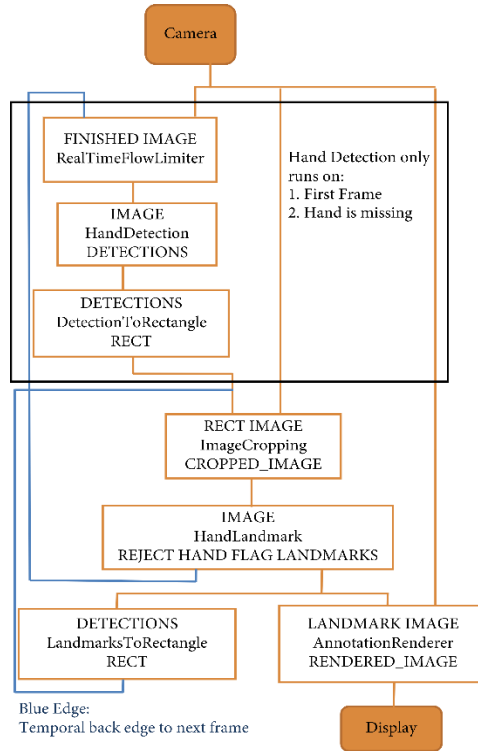
Figure 1: MediaPipe hand recognition graph

2. OpenCV:

OpenCV is a computer vision library which contains image-processing algorithms for object detection. OpenCV is a library of python programming language, and real-time computer vision applications can be developed by using the computer vision library. The OpenCV library is used in image and video processing and also analysis such as face detection and object detection.

3. Single-shot detector model:

Single-shot detector model is used for detecting and recognizing a hand or palm in real time. The single-shot detector model is used by the MediaPipe. First, in the hand detection module, it is first trained for a palm detection model because it is easier to train palms. Furthermore, the non-maximum suppression works significantly better on small objects such as palms or fists. A model of hand landmark consists of locating 21 joint or knuckle co-ordinates in the hand region, as shown in Figure 2.
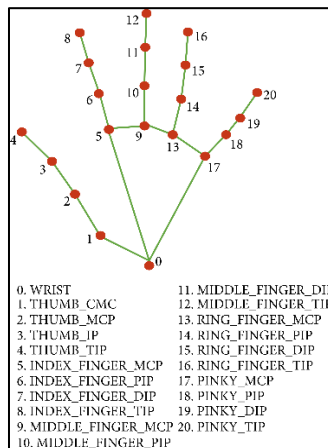

Figure 2: Co-ordinates or land marks in the hand

4.Methodology:

The various functions and conditions used in the system are explained in the flowchart of the real-time AI virtual mouse system in Figure 3.
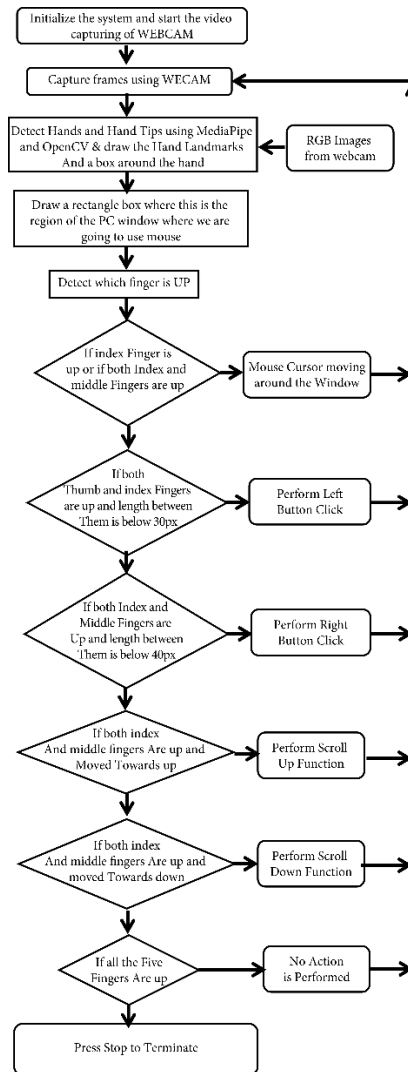

Figure 3: Flowchart of the real-time AI virtual mouse system.

5.1. The Camera Used in the AI Virtual Mouse System

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.

5.2. 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 as shown in the following code:

```
def findHands(self, img, draw = True):
imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
self.results = self.hands.process(imgRGB)
```

5.3. (Virtual Screen Matching) Rectangular Region for Moving through the Window

The AI virtual mouse system makes use of the transformational algorithm, and it converts the co-ordinates of 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.

5.4. Detecting Which Finger Is Up and Performing the Particular Mouse Function

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

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

5.5.1. 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.

5.5.2. 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 click using the pynput Python package.

5.5.3. 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 lesser than 40 px, the computer is made to perform the right mouse button click using the pynput Python package.

5.5.4. 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 40 px and if the two fingers are moved up the page, the computer is made to perform the scroll up mouse function using the PyAutoGUI Python package.

5.5.5. For the Mouse to Perform Scroll down 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 down the page, the computer is made to perform the scroll down mouse function using the PyAutoGUI Python package.

5.5.6. For No Action to be Performed on the Screen

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

IV.   CONCLUSION

The main objective of the AI virtual mouse system is to control the mouse cursor functions by using the hand gestures instead of using a physical mouse. The proposed system can be achieved by using a webcam or a built-in camera which detects the hand gestures and hand tip and processes these frames to perform the particular mouse functions.
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.

The model has some limitations such as small decrease in accuracy in right click mouse function and some difficulties in clicking and dragging to select the text. Hence, we will work next to overcome these limitations by improving the finger tip detection algorithm to produce more accurate results.

V.   APPLICATIONS

The AI virtual mouse system is useful for many applications; it can be used to reduce the space 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.

Major applications:

[1] 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.
[2] 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.
[3] The system can be used to control robots and automation systems without the usage of devices(iv)2D and 3D images can be drawn using the AI virtual system using the hand gestures.
[4] AI virtual mouse can be used to play virtual reality- and augmented reality-based games without the wireless or wired mouse devices.
[5] Persons with problems in their hands can use this system to control the mouse functions in the computer
[6] In the field of robotics, the proposed system like HCI can be used for controlling robots.
[7] In designing and architecture, the proposed system can be used for designing virtually for prototyping.

REFERENCES

[1] J. Katona, "A review of human–computer interaction and virtual reality research fields in cognitive InfoCommunications," *Applied Sciences*, vol. 11, no. 6, p. 2646, 2021.
View at: Publisher Site | Google Scholar
[2] D. L. Quam, "Gesture recognition with a DataGlove," *IEEE Conference on Aerospace and Electronics*, vol. 2, pp. 755–760, 1990.
View at: Publisher Site | Google Scholar
[3] 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.
View at: Publisher Site | Google Scholar
[4] S. U. Dudhane, "Cursor control system using hand gesture recognition," *IJARCCE*, vol. 2, no. 5, 2013.
View at: Google Scholar
[5] K. P. Vinay, "Cursor control using hand gestures," *International Journal of Critical Accounting*, vol. 0975–8887, 2016.View at: Google Scholar
[6] L. Thomas, "Virtual mouse using hand gesture," *International Research Journal of Engineering and Technology (IRJET*, vol. 5, no. 4, 2018.
View at: Google Scholar