



# INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)  
Dundigal , Hyderabad-500043

## Research Based Learning (Prototype / Design Building) External Evaluation Report

**Title of your Idea** : Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision

**Thrust Area / Sector** : Artificial Intelligence

**Branch** : ECE-B

**Year / Semester** : 3<sup>rd</sup> Year/6<sup>th</sup> Semester

S. No	Name of the Student	Roll Number	Mobile Number	Signature
1	KAKASANI SAITEJA	19951A04D3	8121947959	

## 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 is 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 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. Based on the hand gestures, the computer can be controlled virtually and can perform left click, right click, scrolling functions, and computer cursor function without the use of the physical mouse. The algorithm is based on deep learning for detecting the hands. Hence, the proposed system will avoid COVID-19 spread by eliminating the human intervention and dependency of devices to control the computer.

## INTRODUCTION

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. This paper proposes an 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. Electronic Sensors: Another advanced method is placing some loop detectors or proximity sensors on the road. This sensor gives data about the traffic on the road. According to the sensor data, the traffic signals are controlled.

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.

## **LITRATURE REVIEW**

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.

In 1990, Quam introduced an early hardware-based system; in this system, the user should wear a DataGlove [2]. The proposed system by Quam although gives results of higher accuracy, but it is difficult to perform some of the gesture controls using the system.

Dung-Hua Liou, ChenChiung Hsieh, and David Lee in 2010 [3] proposed a study on “A Real-Time Hand Gesture Recognition System Using Motion History Image.” The main limitation of this model is more complicated hand gestures.

Monika B. Gandhi, Sneha U. Dudhane, and Ashwini M. Patil in 2013 [4] proposed a study on “Cursor Control System Using Hand Gesture Recognition.” In this work, the limitation is stored frames are needed to be processed for hand segmentation and skin pixel detection.

Vinay Kr. Pasi, Saurabh Singh, and Pooja Kumari in 2016 [5] proposed “Cursor Control using Hand Gestures” in the IJCA Journal. The system proposes the different bands to perform different functions of the mouse. The limitation is it depends on various colors to perform mouse functions.

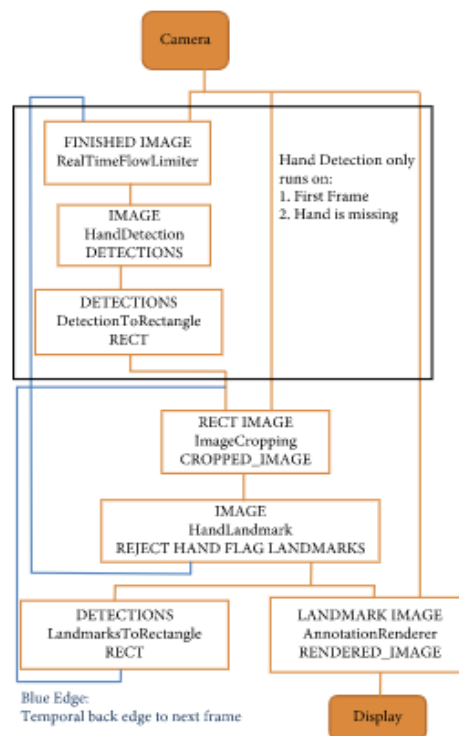
Chaithanya C, Lisho Thomas, Naveen Wilson, and Abhilash SS in 2018 [6] proposed “Virtual Mouse Using Hand Gesture” where the model detection is based on colors. But, only few mouse functions are performed.

## **PROPOSED METHOD**

For the purpose of detection of hand gestures and hand tracking, the MediaPipe framework is used, and OpenCV library is used for computer vision. The algorithm makes use of the machine learning concepts to track and recognize the hand gestures and hand tip.

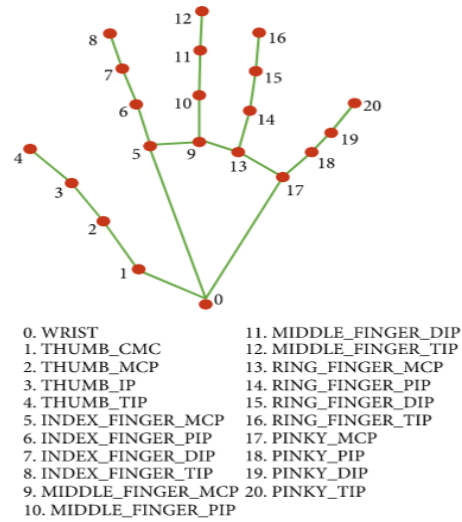
## MediaPipe

MediaPipe is a framework which is used for applying in a machine learning pipeline, and it is an open source framework of Google. The MediaPipe framework is useful for cross platform development since the framework is built using the time series data. The MediaPipe framework is multimodal, where this framework can be applied to various audios and videos. The MediaPipe framework is used by the developer for building and analyzing the systems through graphs, and it also been 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 three 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 which 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.

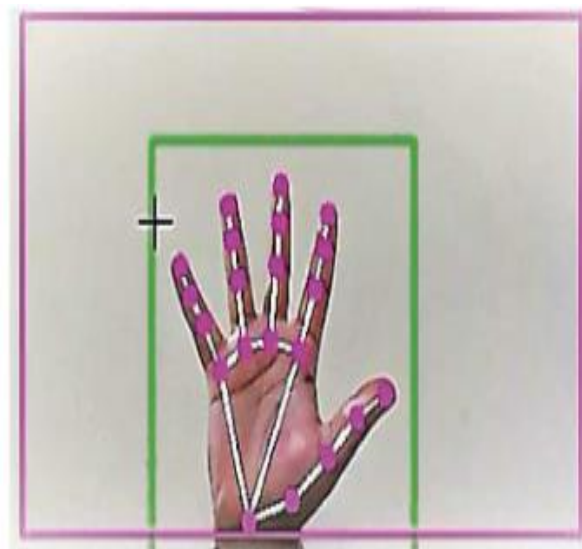


Single-shot detector model is used for detecting and recognizing a hand or palm in real time. The single-shot detector model is used by the MediaPipe. First, in the hand detection module, it is first trained for a palm detection model because it is easier to train palms. Furthermore, the non maximum 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.

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.



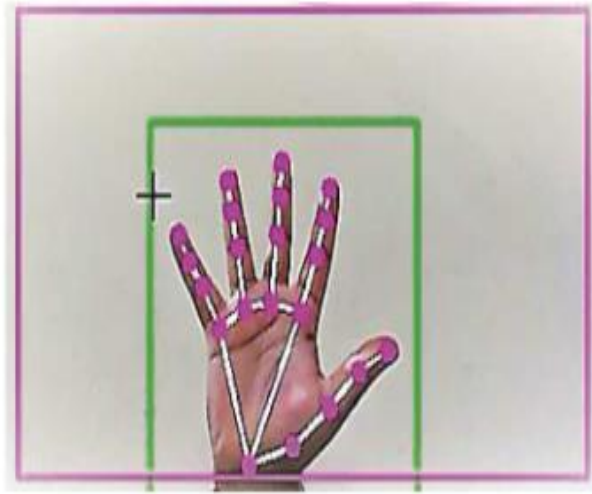
The proposed AI virtual mouse system is based on the frames that have been captured by the webcam in a laptop or PC. By using the Python computer vision library OpenCV, the video capture object is created and the web camera will start capturing video, as shown in Figure 4. The web camera captures and passes the frames to the AI virtual system.



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:

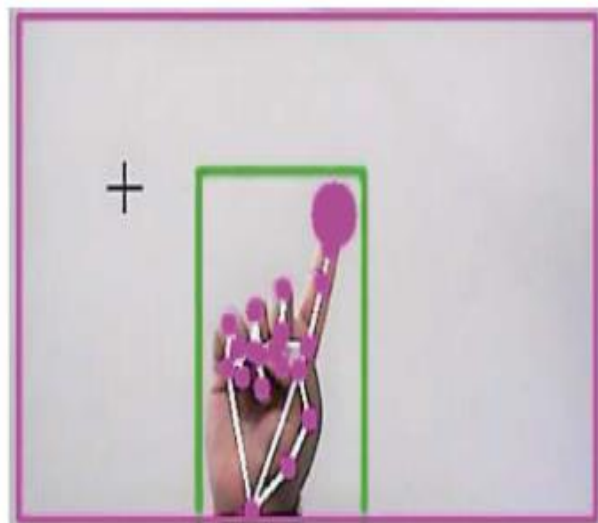
```
def find Hands(self, img, draw = True):
imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
self.results = self.hands.process(imgRGB)
```

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.



### **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.



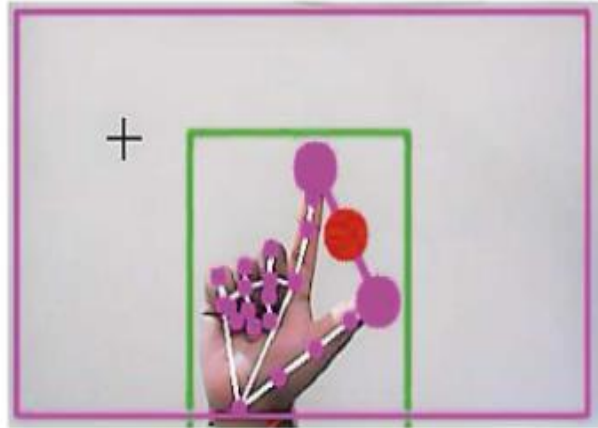
### **For the Mouse Cursor Moving around the Computer Window**

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.



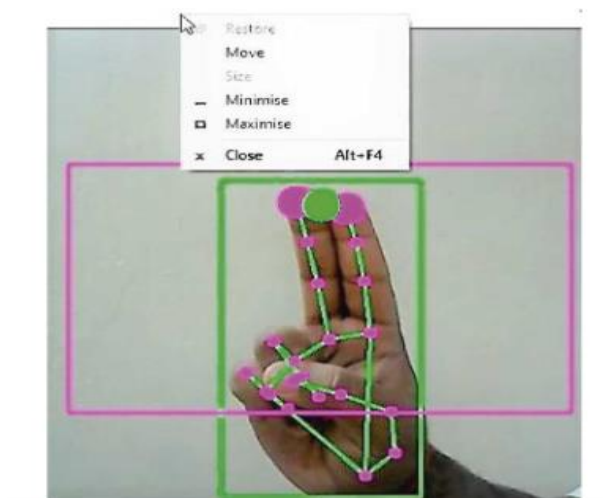
### For the Mouse to Perform Left Button Click

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.



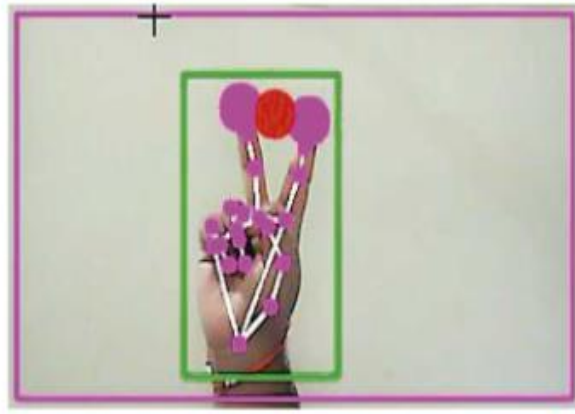
### For the Mouse to Perform Right Button Click

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.



### 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.



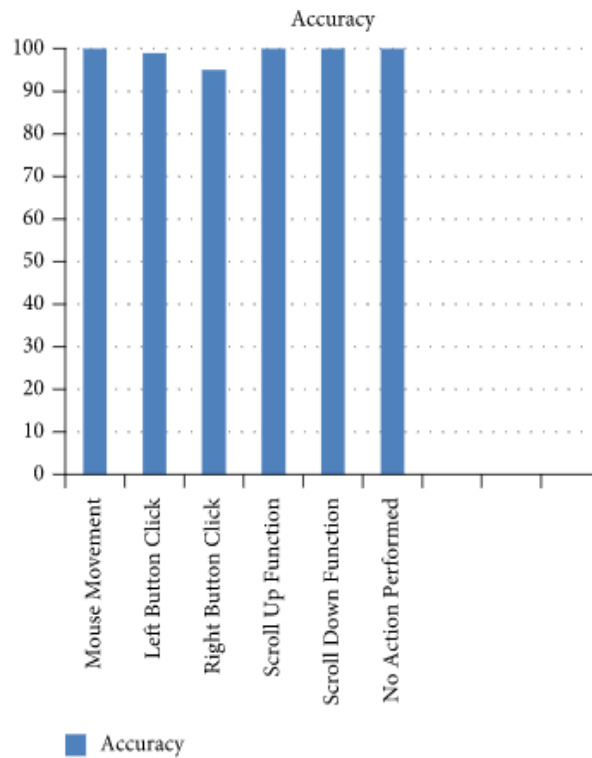
## RESULTS AND ANALYSIS

In the proposed AI virtual mouse system, the concept of advancing the human-computer interaction using computer vision is given.

Cross comparison of the testing of the AI virtual mouse system is difficult because only limited numbers of datasets are available. The hand gestures and fingertip detection have been tested in various illumination conditions and also been tested with different distances from the webcam for tracking of the hand gesture and hand tip detection. An experimental test has been conducted to summarize the results shown in Table 1. The test was performed 25 times by 4 persons resulting in 600 gestures with manual labelling, and this test has been made in different light conditions and at different distances from the screen, and each person tested the AI virtual mouse system 10 times in normal light conditions, 5 times in faint light conditions, 5 times in close distance from the webcam, and 5 times in long distance from the webcam, and the experimental results are tabulated.

Hand tip gesture*	Mouse function performed	Success	Failure	Accuracy (%)
Tip ID 1 or both tip IDs 1 and 2 are up	Mouse movement	100	0	100
Tip IDs 0 and 1 are up and the distance between the fingers is <30	Left button click	99	1	99
Tip IDs 1 and 2 are up and the distance between the fingers is <40	Right button click	95	5	95
Tip IDs 1 and 2 are up and the distance between the fingers is >40 and both fingers are moved up the page	Scroll up function	100	0	100
Tip IDs 1 and 2 are up and the distance between the fingers is >40 and both fingers are moved down the page	Scroll down function	100	0	100
All five tip IDs 0, 1, 2, 3, and 4 are up	No action performed	100	0	100
Result		594	6	99

It can be seen that the proposed AI virtual mouse system had achieved an accuracy of about 99%. From this 99% accuracy of the proposed AI virtual mouse system, we come to know that the system has performed well. The accuracy is low for “Right Click” as this is the hardest gesture for the computer to understand. The accuracy for right click is low because the gesture used for performing the particular mouse function is harder. Also, the accuracy is very good and high for all the other gestures. Compared to previous approaches for virtual mouse, our model worked very well with 99% accuracy. The graph of accuracy;



## 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.

## FUTURE SCOPE

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 be 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).

## 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](#)
7. P. Nandhini, J. Jaya, and J. George, "Computer vision system for food quality evaluation—a review," in *Proceedings of the 2013 International Conference on Current Trends in Engineering and Technology (ICCTET)*, pp. 85–87, Coimbatore, India, July 2013. View at: [Publisher Site](#) | [Google Scholar](#)
8. J. Jaya and K. Thanushkodi, "Implementation of certain system for medical image diagnosis," *European Journal of Scientific Research*, vol. 53, no. 4, pp. 561–567, 2011. View at: [Google Scholar](#)
9. P. Nandhini and J. Jaya, "Image segmentation for food quality evaluation using computer vision system," *International Journal of Engineering Research and Applications*, vol. 4, no. 2, pp. 1–3, 2014. View at: [Google Scholar](#)
10. J. Jaya and K. Thanushkodi, "Implementation of classification system for medical images," *European Journal of Scientific Research*, vol. 53, no. 4, pp. 561–569, 2011. View at: [Google Scholar](#)
11. J. T. Camillo Lugaresi, "MediaPipe: A Framework for Building Perception Pipelines," 2019, <https://arxiv.org/abs/1906.08172>. View at: [Google Scholar](#)
12. Google, MP, <https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html>.
13. V. Bazarevsky and G. R. Fan Zhang. On-Device, MediaPipe for Real-Time Hand Tracking.
14. K. Pulli, A. Baksheev, K. Korniyakov, and V. Eruhimov, "Realtime computer vision with openCV," *Queue*, vol. 10, no. 4, pp. 40–56, 2012. View at: [Publisher Site](#) | [Google Scholar](#)
15. <https://www.tutorialspoint.com/opencv>.

