

# VIRTUSOFT- The Virtual System

Prof. Mangesh Nichat<sup>1</sup>, Masum M. Kajidwar<sup>2</sup>, Ashwini M. Kasture<sup>2</sup>,  
Yash M. Thate<sup>4</sup>, Kanhaiya R. Ade<sup>5</sup>

Guide, Department of Computer Science and Engineering<sup>1</sup>

Degree Students, Department of Computer Science and Engineering<sup>2,3,4,5</sup>

P. R. Pote (Patil) College of Engineering and Management, Amravati, Maharashtra, India

**Abstract:** Many modules are developed to assist the physical world interact with the digital world. Here we present a very unique approach for Human Computer Interaction (HCI) where, we control Cursor movement, Keyboard typing with alternative input hardware collectively switch, and a Camera method presented integrates tricks operation like Clicking and Saving the photos in gallery are visiting be performed with the numerous hand gestures by using webcam as data device employing a real-time web camera. During this "VIRTUSOFT" Our method is to use a web camera and computer vision technology, like image segmentation, background subtraction and color tracking to manage mouse tasks (left clicking, right clicking, double clicking and scrolling actions) which we show how it can perform everything as current hardware devices can. For the thing recognition and tracking the color pointer has been used and implement the module with none physical contact with the system. Click events of the mouse are achieved by detecting the quantity of recommendations on the pictures.

**Keywords:** HCI (Human-Computer Interaction), HSV (Hue Saturation Value), Hand Gesturing, Color Detection, Gesture Recognition, tricks, Layout's, Motion pictures production, Image generation, Cultural differences, Rendering

## I. INTRODUCTION

With the event technologies within the areas of augmented reality and devices that we use in our lifestyle, these devices have become compact within the kind of Bluetooth or wireless technologies. This report proposes an AI VirtuSoft system that produces use of the hand gestures and hand tip detection for performing functions within the computer using computer vision. The most objective of the proposed system is to perform electronic device cursor, keyboard and camera functions employing a web camera or a built-in camera within the computer rather than employing a traditional device. Hand gesture and hand tip detection by using computer vision is employed as a HCI with the pc. With the utilization of the VirtuSoft system, we are able to track the fingertip of the hand gesture by employing a built-in camera or web camera and perform the operations and functions.

While employing a wireless or a Bluetooth mouse, some devices like the mouse, the dongle to attach to the PC, and also, electric battery to power the mouse to control are used, but during this project, the user uses his/her built-in camera or a webcam and uses his/her hand gestures to manage the pc mouse operations. Within the proposed system, the net camera captures and so processes the frames that are captured so recognizes the varied hand gestures and hand tip gestures so performs the actual mouse function.

Python programming language is employed for developing the AI Virtusoft system, and also, OpenCV which is that the library for computer vision is employed within the AI system. within the proposed AI virtual mouse system, the model makes use of the MediaPipe 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 on the move the screen of the pc for performing functions like left click, right click, and scrolling functions. The results of the proposed model showed very high accuracy level, and therefore the proposed model can work okay in realworld application with the employment of a CPU without the employment of a GPU.

**II. PROPOSED SYSTEM**

The proposed system is the Virtusoft System. The Virtusoft system was designed for any laptop to exchange the prevailing manual paper-based system. Currently, the manual system operates on a platform consisting of a straightforward mouse operation employing a hand recognition system that enables the virtual mouse system to perform basic mouse operations like mouse pointer control, left-click, right-click, and drag. A user-friendly keyboard interface is displayed on the monitor that recognizes hands using color segmentation employing a hand recognition system that may recognize hand gestures that determine these characteristics and a flick input method through the camera. Color pixels are obtained by extracting a particular HSV (hue, saturation, value) and applying a threshold mask to the input image. The application process can start when the user's gesture is captured in real time by the webcam. This process processes the captured image for segmentation and identifies pixel values that are adequate to the defined color values. When the segmentation is complete, the whole image is converted to a binary image and also the identified pixels detect the gesture. The position of the white part within the image is recorded and set because the pointer position, simulating the pointer without the utilization of a physical computer. Software features are coded using Python artificial language code and integrate an external library that performs image processing called OpenCV

**III. SYSTEM ARCHITECTURE**

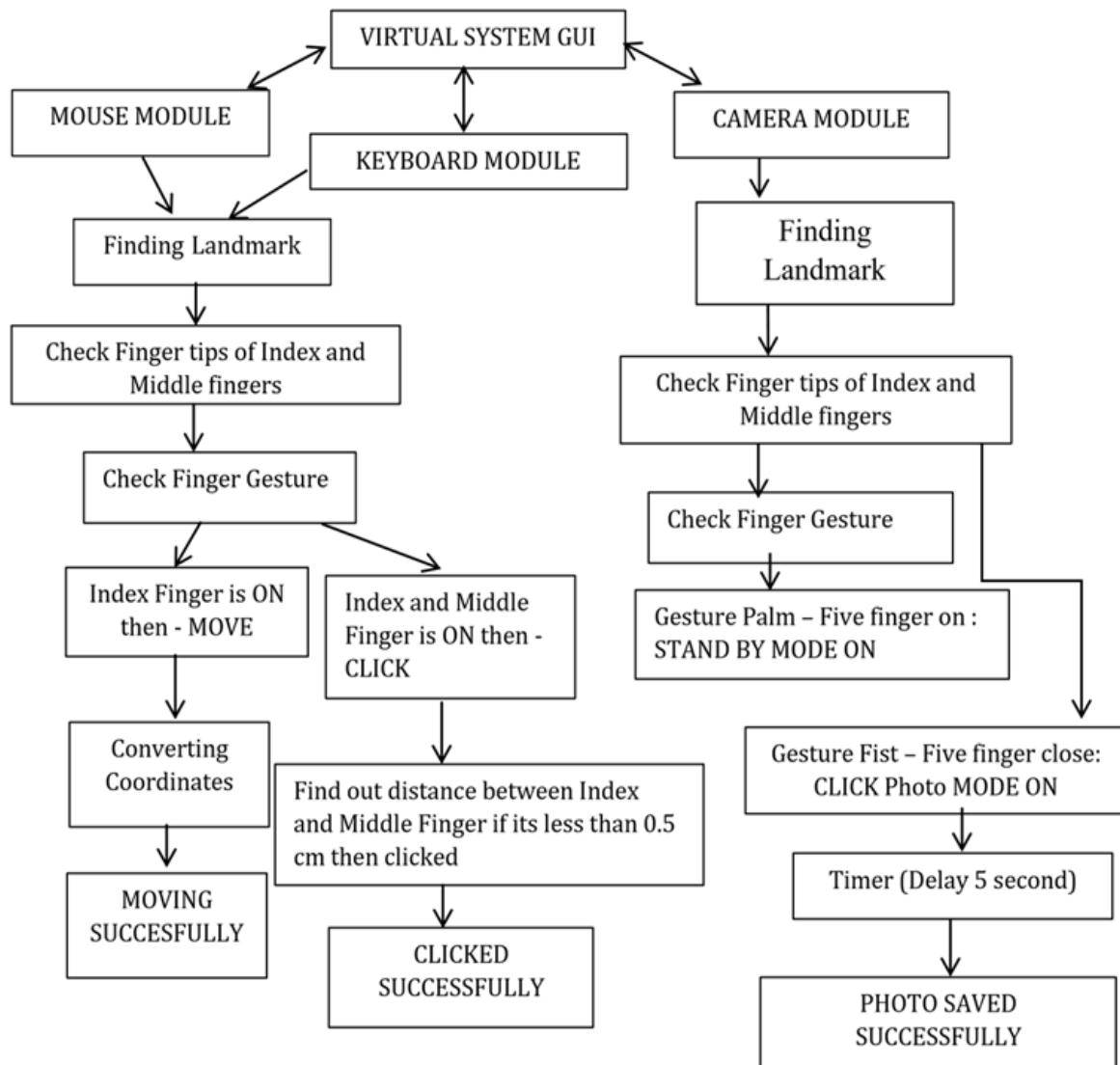


Figure 3.1 below shows the architecture of the virtual AI system "VirtuSoft". In these, we created three modules: a virtual camera, virtual keyboard, and virtual mouse. We used Gesture Recognition to display the output successfully. As you can see, it used the timer to click the photo. In the mouse module, when the camera captures the input of a real-time image, the first camera model detects the hand marks in the input image. Then go to find the fingertip, of the index and middle finger. The camera checks finger gestures and sets conditions as a moving function. After that convert the coordinate pixels to a histogram, find the correct gesture adjustment, and then move the cursor normally. When the index and middle finger are on, it goes to the click option, then the module finds the distance between these two fingers and sets the condition, if the distance is less than 0.5 cm, the photo clicks Will be given to save to gallery and output successfully.

As shown in Figure 3.2.1, the keyboard module first captures the hand landmark in a real-time image after determining the fingertip model to acquire the index and little finger gestures through gesture recognition. After checking if the index finger is turned on, transform the pixel coordinates and switch to navigation mode. When you turn on the index and middle finger, the distance between them is detected and you click the button as gets the output.

As shown in Figure 3.2.1, the camera module first looks for the hand landmark, then the fingertips as the index and middle finger. When all five fingers are on, the palm gesture goes to Standby on mode. When all five fingers are closed, i.e. Fist gesture is used to set the click of the image. When you receive a composition gesture, it times out with a delay of 5 seconds and the photo is saved in the gallery.

#### IV. SYSTEM IMPLEMENTATION

The project will be able to run on all the PC's and compatible with all standard devices. However, for all the development of the project the PC with higher configuration was used. Therefore, the project has the following minimum required system specification.

##### 4.1 Hardware Specification:

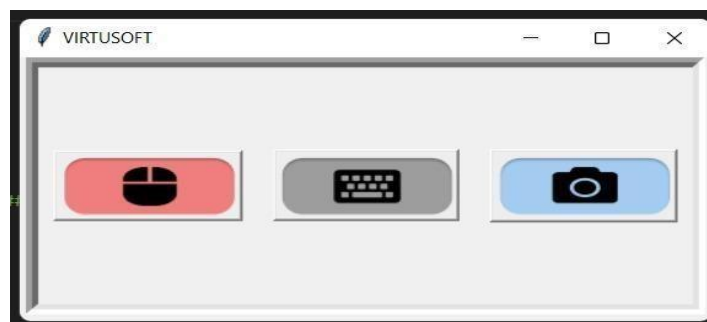
- Processor: Intel i5 Processor
- Resolution: Any Type of Monitor
- RAM: 256 MB or higher
- Disk Space: 10MB (approximately)
- Languages: Python 3.8.8.
- Web-CAM:
- Screen Type: IPS Panels, LCD, LED

##### 4.2 Software Required

- Operating System: Windows 10
- Software: Mediapipe, Tkinter, OpenCV, Numpy, Matplotlib

##### 4.3 System Execution Details

###### Homepage

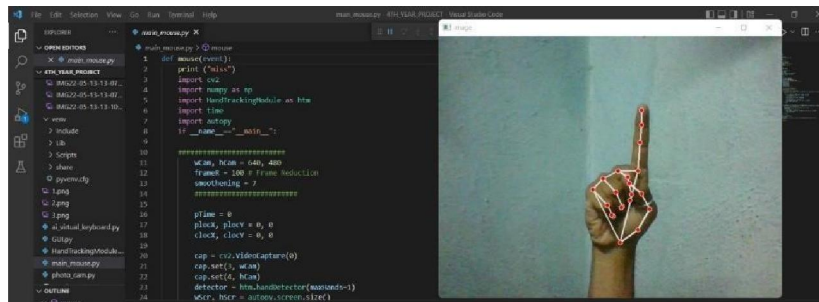


**FIGURE 4.1:** VirtuSoft – The virtual system

The Figure 4.1 above shows you the Homepage of the system. There are Three moduls in this :

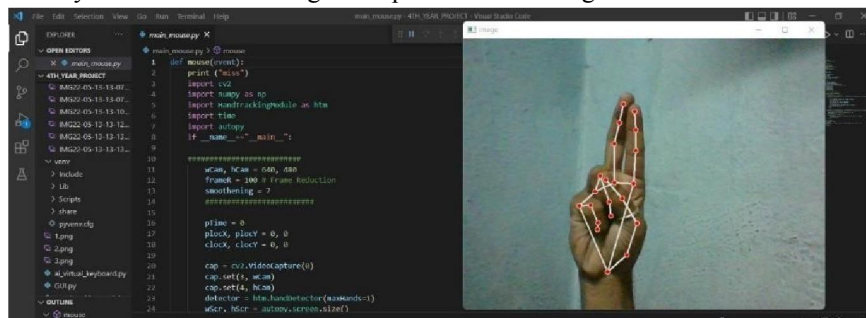
- MOUSE MODULE
- KEYBOARD MODULE
- CAMERA MODULE

**Mouse Model:**



**FIGURE 4.2:** Gesture for dragging the mouse cursor

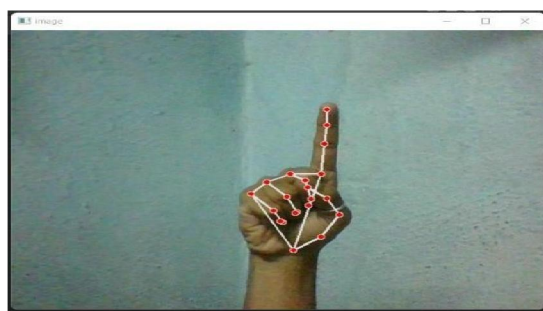
The above Figure 4.2 shows the gesture for Dragging the Mouse Cursor. We used single hand for performing our Hand Gesture module. In these you can see Index Finger is Up and all other fingers are down.



**FIGURE 4.3** Gesture for clicking function

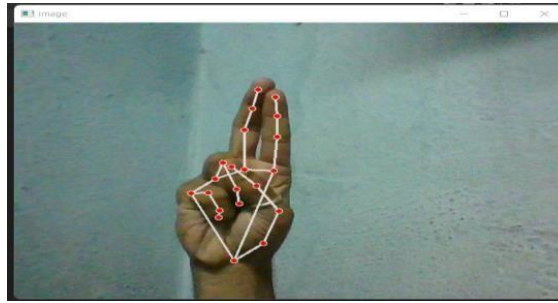
The above Figure 4.3 shows the gesture for Clicking the function. In these you can see Index and Middle finger is up and rest of all is down.

**Keyboard Model**



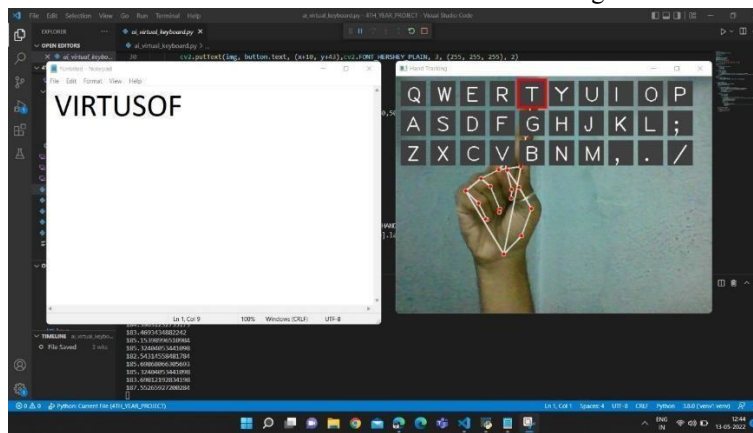
**FIGURE 4.4:** Gesture for navigating keyboard buttons

The above Figure 4.4 shows the gesture for Navigating the Buttons for Virtual Keyboard. Here you can see Index Finger is Up and rest of the fingers are down. These is used for dragging the cursor as well as navigating the buttons.



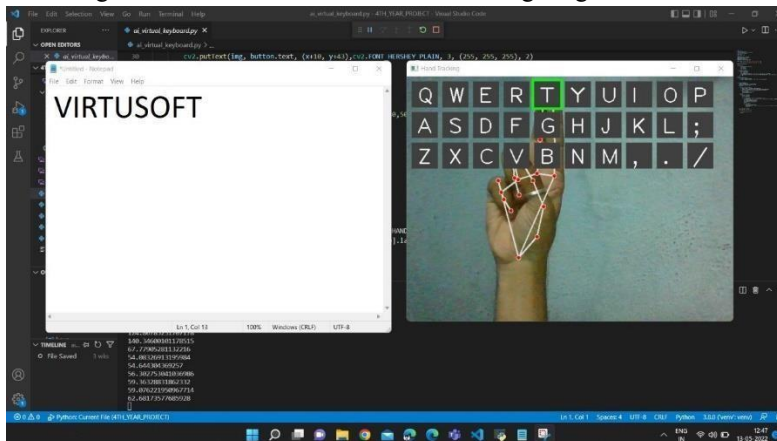
**FIGURE 4.5:** Gesture to select desire button

Above Figure 4.5 shows you the Gesture for Select the Desire button. In these you can see Index and Middle Finger is used and rest of them are down. These is also used in Mouse Model for Clicking the function.



**FIGURE 4.6:** Gesture for navigating letter “ t ”

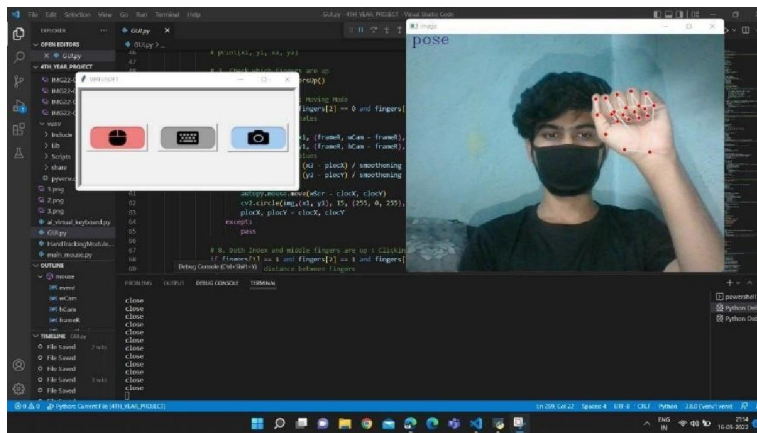
For Example: You can see in Figure 4.3.6 Here Gesture is used for navigating the Letter“ T ”.



**FIGURE 4.7 :**Gesture for typing letters

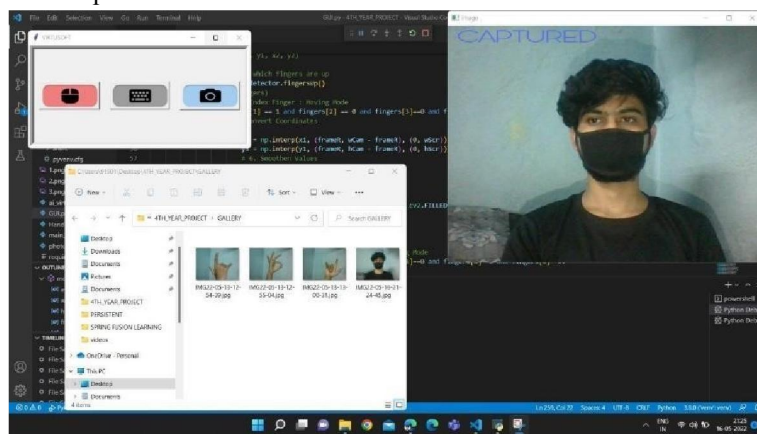
In these Figure 4.7 we can observe that letter “ T ” is typed on notepad in front of VIRTUSOF.

**Camera Model**



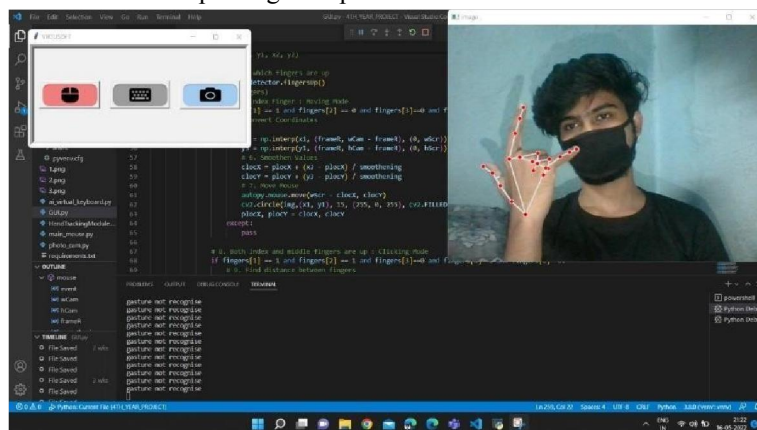
**FIGURE 4.8** Close gesture for instruct camera for posing

Above Figure 4.8 shows Close Gesture which is used to instruct the computer we are going to give Pose for Clicking the Photo, there is POSE word appear in left of the corner. After giving these pose there will be timer process which includes 5 Seconds for click the photo.



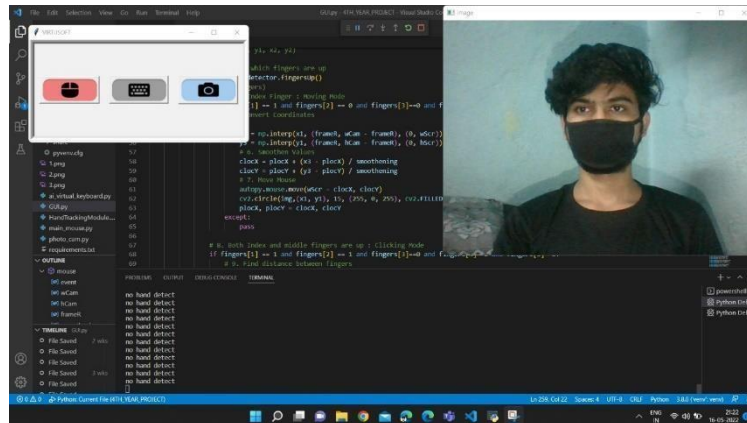
**FIGURE 4.9** posture saved in gallery

Here in Figure 4.9 we can see Photo is Captured and also Saved in the Gallery and CAPTURED word appears on the top left corner of camera to know that our photo gets captured.



**FIGURE 4.10:** No other gesture recognized

The gesture other than “Close Gesture” is not recognized in our camera. If we do any other Gesture these will show us “GESTURE NOT RECOGNISED” in our terminal as we can see in Figure 4.10



**FIGURE 4.11:** No hand detector

Terminal also gives us “NO HAND DETECTED” as a output if we don’t give any gesture related to program of input to our VIRTUSOFT camera which is shown in Figure 4.11

**V. RESULTS AND DISCUSSION**

The proposed virtual AI system is given the concept of using computer vision to further develop humancomputer interaction.

Due to the limited number of datasets available, it is difficult to compare AI virtual system tests to each other. Hand gestures and fingertip recognition were tested in different lighting conditions and at different distances from the webcam to track hand gestures and fingertip recognition. Experimental tests were performed to summarize the results shown in Table 5.1. The test was run by four people and yielded 600 gestures with manual annotations. This test was performed at different lighting conditions and different distances from the screen, and each person tested the VirtuSoft system under normal and low lighting conditions. Table 5.1 shows the test results when the distance from the webcam is short and the distance from the webcam is long.

From Table 5.1, we can see that the proposed virtual AI system achieved about 99% accuracy. This 99% accuracy of the proposed virtual AI system shows that the system worked well. As shown in Table 5.1, the accuracy of "right-click" is low because it is the most difficult gesture for a computer to understand. Rightclicks are less accurate because the gestures used to perform certain mouse functions are more difficult.

The accuracy of all other gestures is also very high and high. Compared to the previous virtual mouse approach, our model worked very well with 99 ° currency. The accuracy chart is shown in Figure 5.1.

Hand Tip Gesture	Function performed	A.T	M.K	P.W	M.A	Overall	Accuracy
Tip ID 1 or both tip IDs 1 and 2 are up	Cursor, Keyboard and Camera Movement	100	100	100	100	100	100
Tip IDs 0 and 1 are up and the distance between the fingers is <30	Left Button Click	100	99	100	99	99	99
Tip IDs 1 and 2 are up and the distance between the fingers is <40	Right Button Click	99	97	99	94	95	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	100	100	99	100	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	99	100	100	100	100

All five tip IDs 0, 1, 2, 3, and 4 are up	No Function Performed	100	100	100	100	100	100
Result						594	99

TABLE NO 5.1: EXPERIMENTAL RESULTS

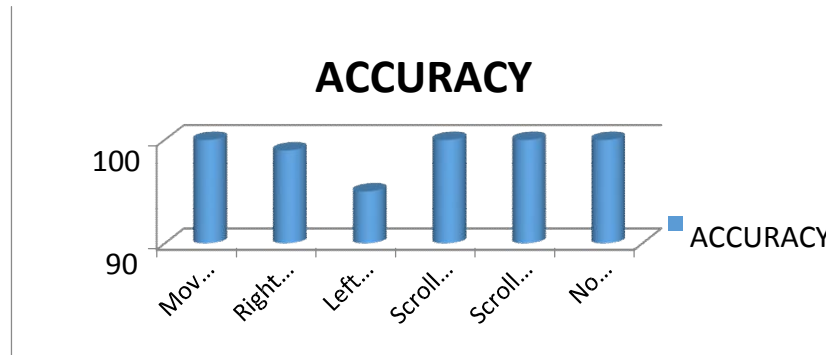


FIGURE 5.1: Accuracy graph

## VI. CONCLUSION

The main goal of AI Virtusoft is to control mouse pointer, keyboard, and camera functions with hand gestures instead of using physical hardware. The proposed system can be achieved using a webcam or builtin camera that recognizes hand gestures and hand tips and processes these frames to perform specific mouse functions. From the model results, we can conclude that the AI proposed by Virtusoft works very well, is more accurate than the existing model, and the model overcomes most of the limitations of the existing system. .. Because the proposed model is more accurate, the proposed system can be used virtually with hand gestures without the traditional physical touch, allowing AI Virtusoft to be used in real-world applications. The model has some limitations, including: The click keyboard function is slightly less accurate and causes problems when clicking and dragging to select text. Therefore, we will then work to overcome these limitations by improving the fingertip detection algorithm to provide more accurate results.

## REFERENCES

- [1]. <https://www.ibm.com/cloud/learn/machine-learning> "Concept of Machine Learning" by IBM Cloud Education.
- [2]. Miika Silfverberg, in "Text Entry Systems", 2007.
- [3]. D. L. Quam, "Gesture recognition with a DataGlove," IEEE Conference on Aerospace and Electronics, vol. 2, pp. 755-760, 1990.
- [4]. K. P. Vinay, "Cursor control using hand gestures" International Journal of Critical Accounting, vol. 0975-8887, 2016
- [5]. L. Thomas, "Virtual mouse using hand gesture" International Research Journal of Engineering and Technology (IRJET), vol. 5, no. 4, 2018.
- [6]. Granit Luzhnica, Elizabeth Lex, Viktoria Pammer. A Sliding Window Approach to "Natural Hand Gesture Recognition using a Custom Data Glove".