International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

AI Virtual Mouse and Keyboard.

Shweta Pardeshi, Shubham Jagtap, Shashank Kathar, Shivam Giri, Prof. Suresh Kapare

Department of Computer Science and Engineering MIT ADT University, Pune, MH-India
Department of Computer Science and Engineering MIT ADT University, Pune, MH-India
Department of Computer Science and Engineering MIT ADT University, Pune, MH-India
Department of Computer Science and Engineering MIT ADT University, Pune, MH-India
Department of Computer Science and Engineering MIT ADT University, Pune, MH-India

Submitted: 15-05-2022 Revised: 25-05-2022 Accepted: 28-05-2022

ABSTRACT
Since the development of computer technology, human interaction with computers has evolved. Gestures are an effective method to interact also the era of Covid-19 impacted on us. The mouse as well as the keyboard are devices that interact with computers. Here we have tried to make the mouse and keyboard feature interaction using hand gestures. Eventually discarding the electronic equipment. Hence, controlling the mouse cursor by using your finger and typing on a virtual keyboard. The actions such as clicking, dragging and typing data will be carried out using various hand gestures. The IOT device needed to achieve this is a webcam. The camera's output will be presented on the system's screen so that the user can further calibrate it. We use technologies like Open-CV, Media-Pipe, Python. The Media-Pipe library comes in very handy in AI projects and provides features that help the model's efficiency. The user will be able to navigate the computer cursor and type using the virtual keyboard with their hand holding color caps or tapes, and left click and dragging will be done using various hand motions. In this paper, we propose a hand gesture recognition system to control the virtual mouse and a virtual keyboard for natural human computer interface.

KEYWORDS-Hand Tracing, Finger counter, AI Virtual Mouse, AI virtual Keyboard, Gesture control.

I. INTRODUCTION

Human computer interaction is one of the most rapidly growing technologies. Hand gestures are very significant for human-computer interaction. Hence discovering various modules of HCI via hand gestures and implementing them to innovate is the objective. The proposed systems consist of projects like Hand Tracking, Finger counting, gesture volume control and finally AI Virtual Mouse as well as Virtual Keyboard. Each are interrelated and are basics to advance. In order to increase productivity and to keep the flexibility in each module creating mini projects is the base idea. Since the development of computer technology, the technique for constructing a process of human-computer interaction has evolved. In terms of human-computer interface, the mouse and keyboard are excellent devices.

The model suggested ultimately discards the electronic equipment. The functionality of mouse and keyboard will be done by human hands. The system requires an input IOT device, webcam. The model suggests detecting the human hand and tracking its gestures. The gestures which include pointing finger, touching the tips of finger result in various functionalities of mouse and keyboard. Further detection of gestures the functionalities are done like opening a notepad application, typing on the notepad application. The camera's output will be presented on the system's screen so that the user can further calibrate it. NumPy and mouse are the Python requirements that will be utilized to create this system-In project phase one implementation and exploration is done on a Virtual Mouse and on Virtual keyboard in project phase two. Mini projects like Hand Tracking which tracks the palm of the hand and displays Frame rate, Finger counting that counts fingers and uses a hand tracking module as its base are also included. Later, Gesture Volume control which controls the volume by extracting certain hand features is also implemented. These are projects that are targeted to increase productivity. We use technologies like Open-CV, Media-Pipe, and Python. Media-Pipe is developed by Google. It is quite efficient and helps to provide quick solutions to AI projects.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

II. LITERATURE SURVEY

Industrial robot control, sign language translation, smart surveillance, lie detection, visual environment manipulation, and rehabilitation devices for those with upper extremity physical limitations can all benefit from virtual hand gesture recognition. In addition to being inconspicuous, virtual hand gesture recognition systems may be a natural means of interacting with machines, making them an important sort of input mechanism. The ability to reliably detect gestures from numerous perspectives is a difficult component of this technology[2]. According to a survey and a sign language study, the hand gesture is the easiest and most natural manner of communicating among the many gesture communications modalities. With the help of recent breakthroughs in the field of computer vision and pattern recognition, real-time vision-based hand gesture identification is becoming more and more possible for Human-Computer Interaction. [4].

For laptops, there are a variety of quick access techniques for hand and mouse gestures. Using our project, we could use the laptop or webcam to recognize the hand motion and operate the mouse and execute simple operations like mouse pointer control, select and deselect using left click. The completed project is a laptop based "Zero Cost" hand recognition system that employs simple algorithms to determine the hand, hand movements, and assign an action to each movement. The system we're building, which is written in Python, is much more responsive and easy to implement because Python is a simple language that is platform independent, flexible, and portable, all of which are desirable qualities in a program aimed at creating a Virtual Mouse and Hand Recognition system[8].

Hand Gesture detects shapes and/or positions based on implementation to task the system to perform some job. The learning process starts with observations or data, such as examples, direct experience, or instruction, in order to find patterns in data and make better decisions in the future based on the examples we provide. The paper's main goal is to allow computers to learn without human involvement or aid and change their activities accordingly. In this system, a semi-supervised machine learning technique is implemented, which can significantly enhance learning accuracy when the collected data requires experienced and relevant resources to train and learn from[11]. The system uses Hand motions recognition system to replace the essential pointing devices used in PCs to depict hand motions. The purpose of this project is to create a solution to a technology platform that mechanically identifies and classifies disease. The detection of disease is accomplished by steps such as picture loading, pre-processing, segmentation, extraction, and classification. The photographs of the leaves are used to identify plant illnesses[12].

The computer's camera analyzes the image of a person's hand making various movements, and the mouse or pointer moves in reaction to the movements, including performing right and left clicks with different gestures. The keyboard may be controlled with a variety of gestures, including a one-finger motion to select an alphabet and a four-figure gesture to swipe left and right [13]. It will work as a virtual mouse and keyboard without the usage of any wires or other devices.

This survey presents a virtual keyboard that overcomes the difficulties faced by using standard computing devices such as computers and laptops. The keyboard is built primarily on a vision-based human-computer interaction idea, picture collecting, and image processing approach that includes virtual keys that are sized to match the dimensions of standard QWERTY keys. Image Capturing, Character Identification, and Device Emulation are the three primary modules of this keyboard[14]. A 3D model-based approach is a technique which has been proposed for hand tracking. A structured light system generates dense 3D data, which is used in the system. To remove the background, we use a static reference image. The 3D hand model is sampled and matched against the depth map in each frame, and the hand model's deformation parameters are tuned to minimize the Euclidean distance between the model and depth map surfaces[16].

The objective of this work is to explore some of the recent research in the hand gesture recognition system and to detect the virtual mouse for human and computer interaction.

III. RESEARCH MODEL

3.1 TECHNOLOGY USED

3.1.1OpenCV:

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision.The library is cross-platform and free for use under the open-source Apache 2 License. Starting in 2011, OpenCV features GPU acceleration for real-time operations.

3.1.1.1 Application of OpenCV:

It is used for image frame acquisition, real time image frame object detection and text modification in window. Also, works quite

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

efficiently in real time applications. Few important functions implemented are

- imshow() - used to display an image in a window.
- putText() - used to draw a text string on any image. In our case we used to display frame rate.
- waitkey() - allows users to display a window for given milliseconds or until any key is pressed. In our case we passed parameter as 1 which initiates a while loop, and the image is displayed continuously giving a video effect.

It was found that OpenCV is easier to implement and debug than MATLAB.

3.1.1.2 Challenges of OpenCV:

We found that the frame rate (fps) should be 30 for the output accuracy. Hence ensuring the same is necessary. Calculating fps for webcam is not done by OpenCV functions (CAP_PROP_FPS); we manually calculated the values by finding previous time and current time.

3.1.2 Web Camera:

A webcam is a video camera that feeds or streams an image or video in real time to or through a computer network, such as the Internet. This IOT device can be an integrated camera of the laptop or an external webcam with high resolution.

3.1.2.1Applications of Webcam:

To get computer vision.

3.1.2.2Challenges of Webcam:

If the camera used is within a laptop or integrated the VideoCapture() function implemented should have parameter 0. If the webcam is an external device the parameter passed should be 1 or else the video will not be detected.

3.1.3 MediaPipe:

MediaPipe offers cross-platform, customizable ML solutions for live and streaming media.[17] It provides machine learning pipelines. This perception pipeline is a graph. Developed by Google. In 2019, MediaPipeopened up a new world of opportunities for researchers and developers to develop applications in the computer vision sector, following its public release.

3.1.3.1 Application of MediaPipe:

MediaPipe is implemented for detecting the palm and drawing out the hand landmarks. It has developed these models which have been implemented. MediaPipe has trained thirty thousand images of palm hence it provides great accuracy. The library detects 21 hand landmarks which can be coordinated to build various functionalities.

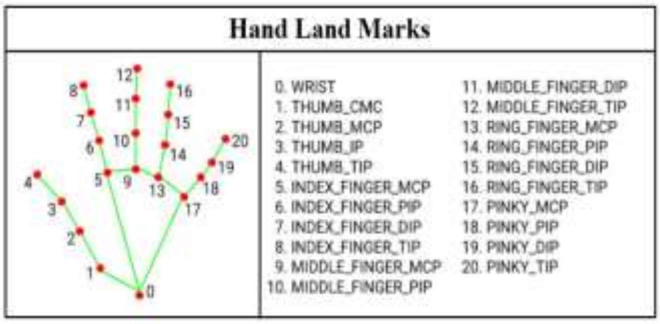

Fig.1. Hand Landmarks by MediaPipe [17]

Code includes a hand class which includes methods. Examples of the following are:

- mp.solutions.hands - initialize the Hands class where MediaPipe is imported as mp.

3.1.3.2Challenges of MediaPipe:

While implementing the MediaPipe library we have to make sure that the image captured in the image frame is in RGB format which by default is obtained as BGR. Hence, we have to convert it.

3.1.4 AutoPy:

It is a simple cross-platform GUI automation library for Python. It includes functions for controlling the keyboard and mouse, finding colors and bitmaps on-screen, and displaying alerts -- all in a cross-platform, efficient, and simple

- Mp.solutions.drawing_utils - responsible to draw all hand's landmarks points on the output image which were detected by our Hands function.

manner. In our module we have used the functions for controlling virtual mouse and keyboard. [18]

3.1.4.1 Application of Autopy:

In the Autopy the mouse module contains functions for getting the current state and controlling the cursor of the mouse virtually.

Few implemented functions include:

- Autopy.mouse.move (x, y) : The function will move the mouse according to the imputed x and y coordinate.

3.1.5 NumPy:

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

Library is used to perform mathematical operations on arrays, matrices as well as linear algebra. This computing library is designed to integrate with Python and can be called the core library for scientific computing in Python. [19]

3.1.5.1 Application of NumPy:

NumPy's computational power is used in our module to find the coordinates of the screen while implementing AI Virtual mouse. The webcam will return the coordinates in dimension (640 * 480) of the fingertip. Hence, converting these coordinates equivalent to the user's screen NumPy's mathematical computation is required.

3.1.6 Time:

Library is used to get the current time. In the proposed system the library is used to calculate the fps. That is the frame rate.

3.1.6.1 Application of time:

Frame rate is calculated as (1 / (cTime- pTime)) where cTime is current time and pTime is previous time.

3.1.7 Pynput:

It is a library used to control input devices. It can be used to control the mouse and keyboard.

3.2 IMPLEMENTATION OF THE MODELS

3.2.1 OVERVIEW:

The aim is to create a cost-free hand recognition system for controlling laptops and PCs with support of an IoT device i.e., webcam. Recognizing the hand gesture, we could control the cursor of the mouse and perform basic operations like mouse pointer or cursor control as well as features of the mouse like left click and right click. The idea also suggests how the fingers are gestured will define the type of click. If only the index finger's tip is detected, then the cursor will act as the finger moves like a mouse. The hand movements are small and easy to do. Hence the system is also designed such that it can be used easily. Similarly, with an AI keyboard the process starts with designing the keys and the entire keyboard onto the screen, hence implementing. The tips of the finger are also used to select the alphabets off the screen resulting in typing on the text area on the screen or on external applications like notepad.

The system also focuses on developing other small feature models such as hand tracking, finger counter and gesture volume control along with the AI virtual mouse. The reason is to increase the results and to get more outcomes from a single project.

3.2.2 MINI MODELS

The mini models include:

- Hand tracking
- Finger counting
- Volume Control

3.2.2.1 Block diagram of Mini models (Hand tracking, finger counting, Gesture Volume control)

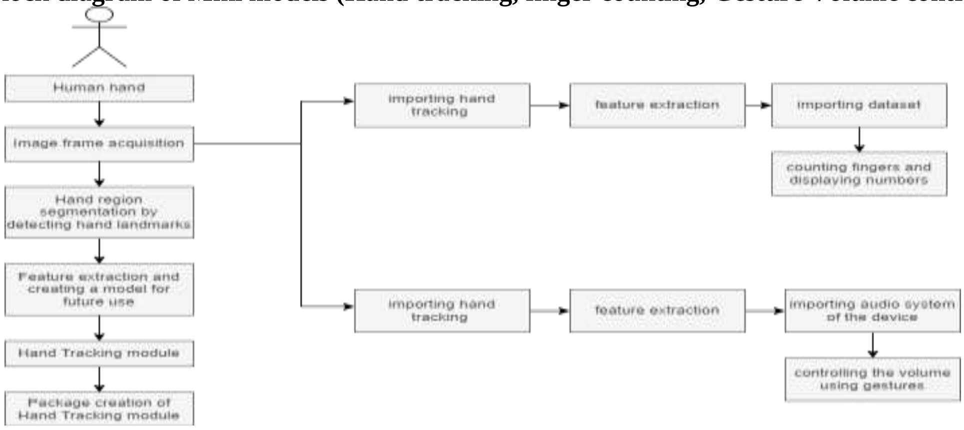

Fig. 2. Block diagram of Mini Models

3.2.2.2 HAND TRACKING - The Hand Tracking model will use two main modules of the MediaPipe in the backend that are palm detection and hand landmarks. The Palm detection crops the image and finds the exact image which has a palm. Hand Landmarks; find 21 different landmarks on the cropped image. Hence the MediaPipe library tracks the hand by segmenting it into a category of 21 features.

- Output obtained: Package created to track hand.
- Flowchart for Hand Tracking Model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

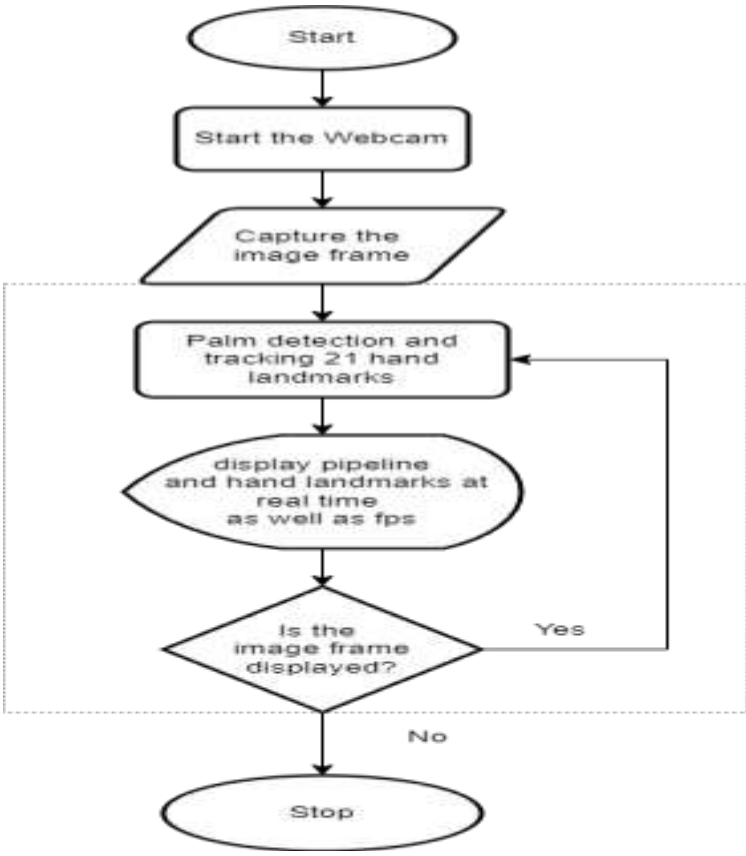

Fig. 3. Flow ofHand tracking Model

3.2.2.3 FINGER COUNTER -Finger Counter will import the Hand tracking model. As the Hand Tracking model will contain modules which are flexible and easing reusable importing and using it will be quite efficient. Finger counter will also import a database that has 6 images of hand counting till five and an image of fist. The model will trace the 21-hand landmark, check their position in real time and then give the estimated output of how many fingers are displayed and then it will count fingers.

- Output obtained: How to detect fingers. How to extract the feature of the landmark and implement it to create a functionality.
- Flowchart for Finger Counter Model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

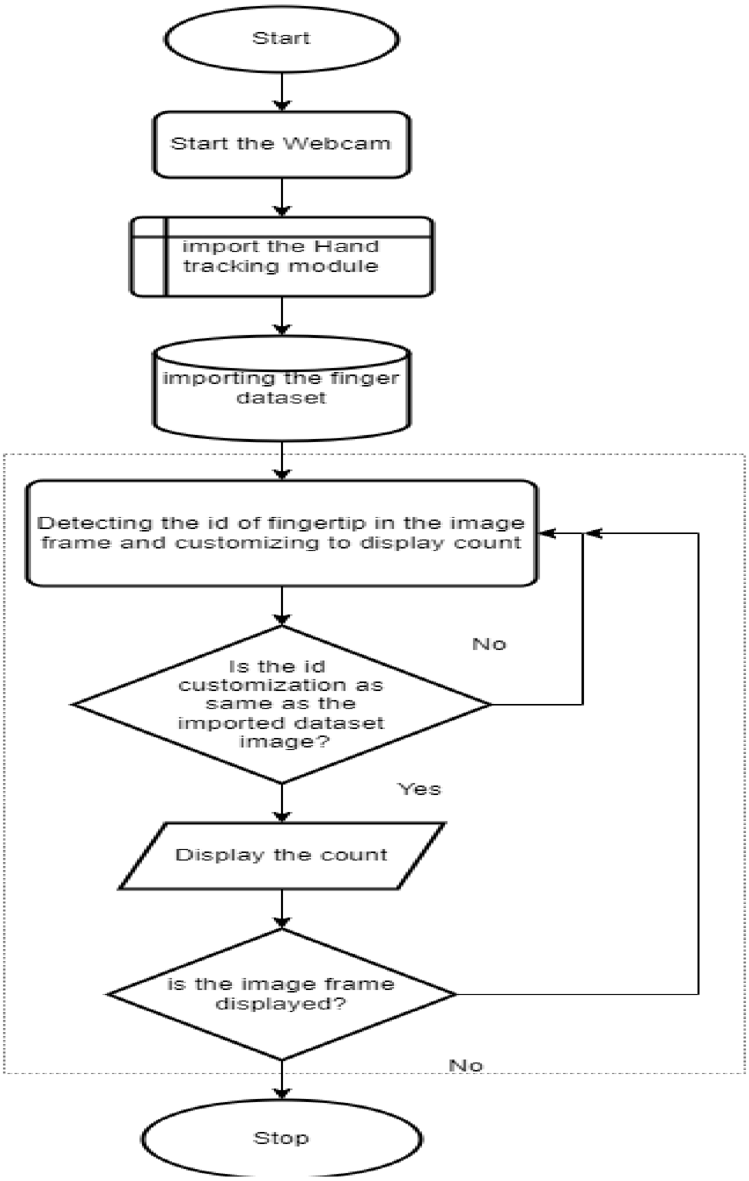

Fig. 4. Flow of Finger Counting Model

3.2.2.4 GESTURE VOLUME CONTROL-This model will also import Hand tracking modules. Detecting index finger and thumb to control the volume. Also detecting the pinky finger to fix the volume. Hence using 21 hand landmark features extracting the landmarks which are required and programming them accordingly to give the expected output is the aim here. Here detecting fingers as well as their gestures was implemented.

- Output obtained: How to create functionalities using gestures was the learning.
- Flowchart for Gesture Volume Control Model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

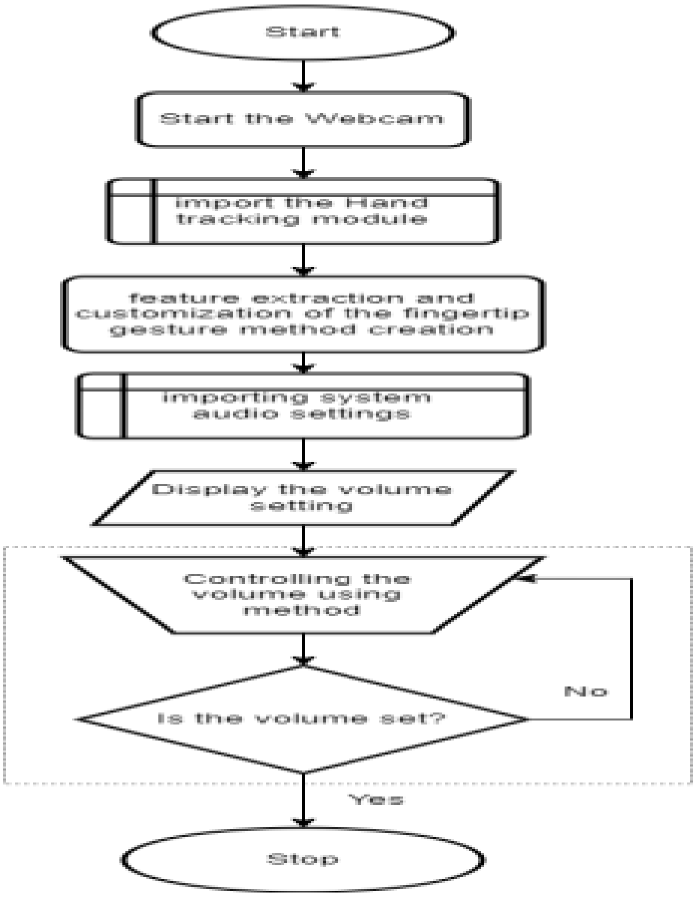

Fig. 5. Flow of Gesture Volume Control Model

3.2.2.5 RESULT OF MINI MODEL

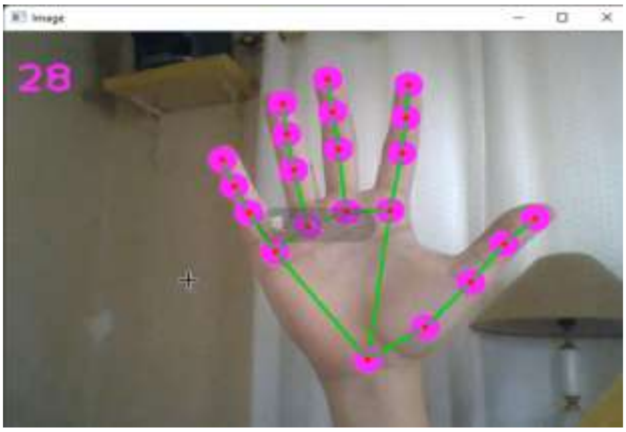

Fig. 6. Result of Hand Tracking Model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

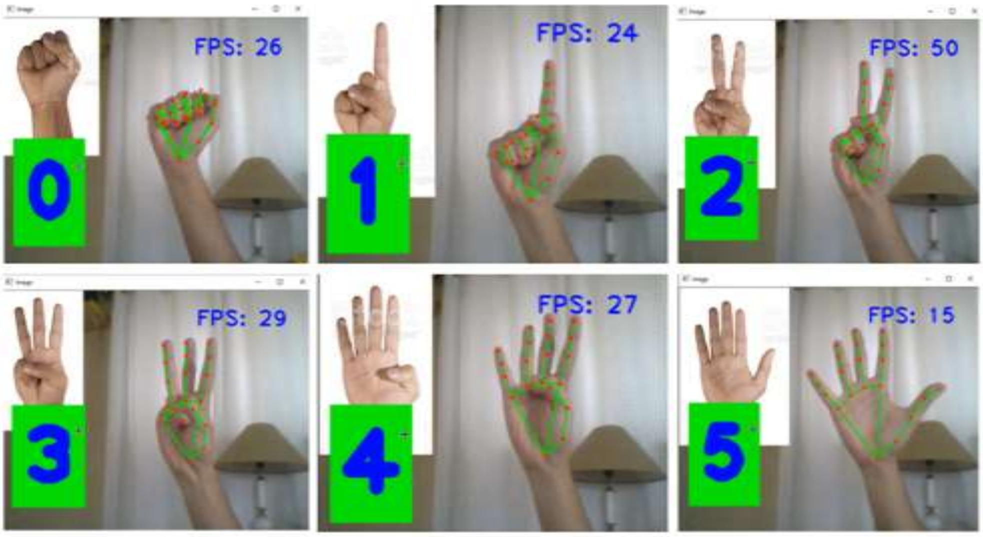

Fig. 7. Result of Finger Counting Model

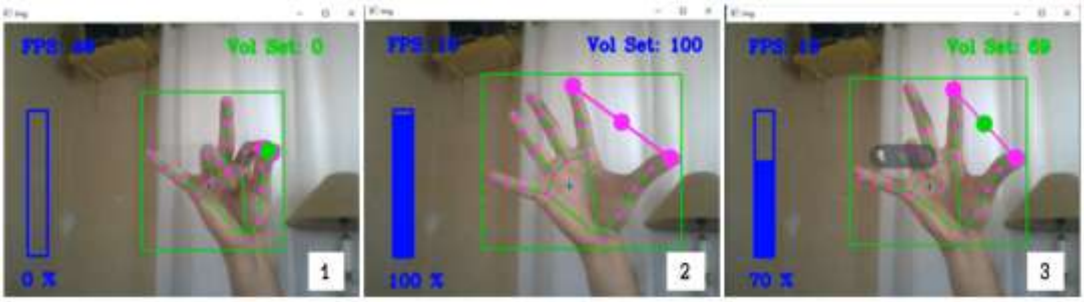

Fig. 8. Result of Finger Counting Model

1) Lowering the volume level to 0%
2) Increasing the volume level to 100%
3) Setting the volume level to 69%

3.2.3 VIRTUAL AI MOUSE:

An AI mouse is a proposed system where the mouse functions will be handled by a finger using the camera. Use of the hand tracking model will be done again. Tracking hand extracting 21-hand landmarks and programming such that only the index finger's movement will be read is the objective. Using the index finger as a pointer to the mouse cursor the virtual mouse is implemented. The right and left click functionality were implemented when index and thumb and index and middle finger touch gestures were detected respectively.

3.2.3.1 FLOW DIAGRAM OF WORKING OF VIRTUAL AI MOUSE

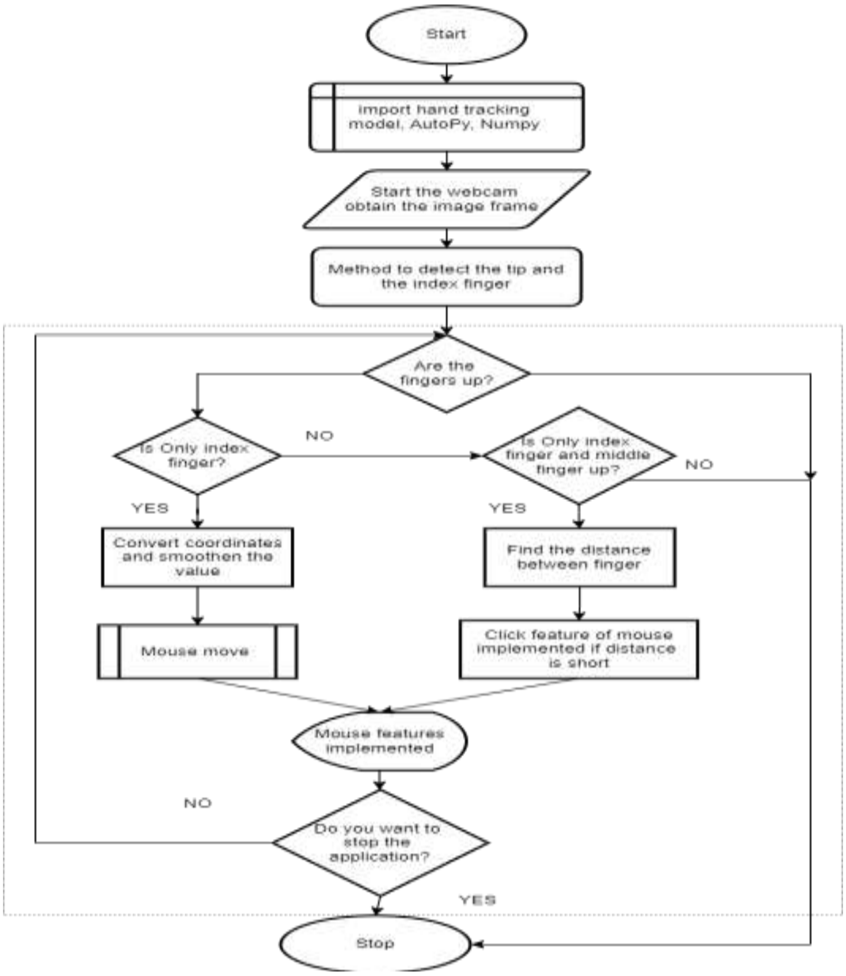

Fig. 9. Flow of Virtual Mouse

3.2.3.2RESULT

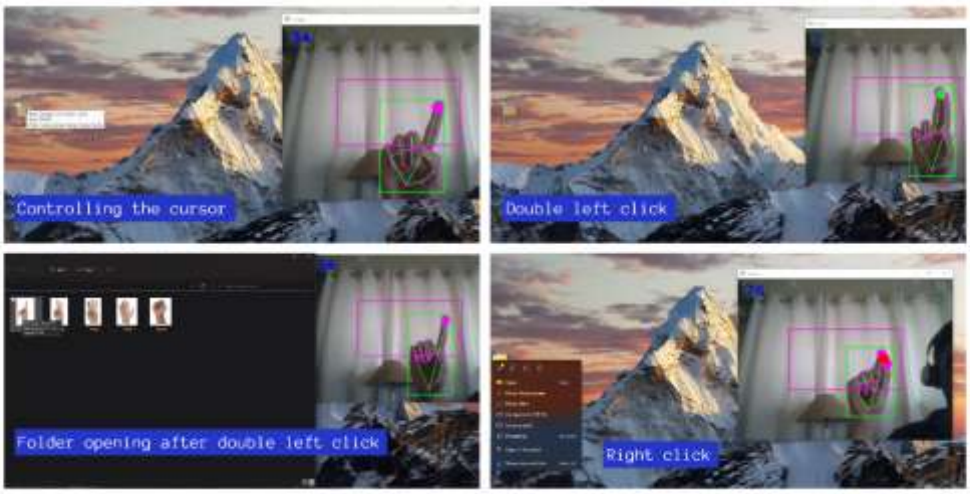

Fig. 10. Result of Visual Mouse

3.2.4 VIRTUAL AI KEYBOARD

An AI keyboard is a proposed system where a virtual keyboard is simulated on the screen for our reference. The letters are detected when a finger hovers over the particular alphabet on the screen. The letter is typed on the gesture of middle

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 5 May 2022, pp: 1925-1936 www.ijaem.net ISSN: 2395-5252

and index finger touch. The AI keyboard implemented is very flexible. As the keyboard is comparatively big the image frame acquisition by web camera has to be adjusted accordingly. The implementation suggests

3.2.4.1 Flowchart of the working of the keyboard.

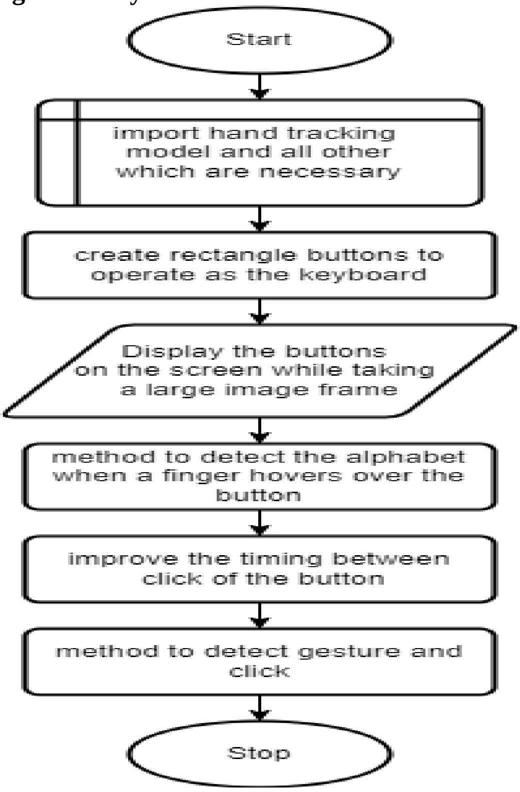

Fig. 11. Flow of Virtual Keyboard

3.2.4.2 RESULT OF VIRTUAL KEYBOARD

Fig. 12. Performing thekeypress on the virtual keyboard

IV. CONCLUSION

Using a real-time camera as well as hand detection and tracking, we have attempted to perform AI Virtual mouse as well as a keyboard. It includes various features like; the mouse includes functions like right click, double left click. The mouse is designed such as it performs like an actual mouse and based on computer vision techniques. The keyboard is simulated as an actual keyboard as well as the alphabets are well aligned. The result includes keypress and key functionalities of space bar and backspace. The system designed can we worked in all conditions just requirement is a bright room and hand as well as a HD webcam for fast detection and better result. The system efficiency will help us in saving space at work, reduce plastic waste by eliminating the keyboard and the mouse. It will also an innovation leading to brighter prospects.

V. FUTURE WORK

System is quiteautonomous. It's one of the best features is that it is easy to use. Future work will be focused on algorithm improvement by merging the models created and making an entire virtual system handler such that it has functionalities of virtual mouse and keyboard as well as it can control the volume, brightness and other functionalities. It also includes improvement in keyboard. We plan to add a caps-lock button. Also, a button that switches the keyboard characters to special characters so that they can be used when they are needed. The virtual keyboard and mouse can be used togetherin airport kiosk check-in machines for touchless check-in experience. The passengers can enter the PNR without touching the screen and get there boarding passes. When such system is implemented.

REFERENCES

[1] Suarez, J., & Murphy, R. R. (2012, September). Hand gesture recognition with depth images: A review. In 2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication (pp. 411-417). IEEE.

[2] Shah, P., Pandya, K., Shah, H., & Gandhi, J. (2019). Survey on vision-based hand gesture recognition. International Journal of Computer Sciences and Engineering, 7, 281-288.

[3] Chen, L., Wang, F., Deng, H., & Ji, K. (2013, December). A survey on hand gesture recognition. In 2013 International conference on computer sciences and applications (pp. 313-316). IEEE.

[4] Panwar, M., & Mehra, P. S. (2011, November). Hand gesture recognition for human computer interaction. In 2011 International Conference on Image Information Processing (pp. 1-7). IEEE.

[5] Cheng, H., Yang, L., & Liu, Z. (2015). Survey on 3D hand gesture recognition. IEEE transactions on circuits and systems for video technology, 26(9), 1659-1673.

[6] Patil, N., Sali, G., &Lokhande, N. (2019). Mouse on Finger Tips using ML and AI.

[7] Sharma, V. K., Kumar, V., Iqbal, M., Tawara, S., &Jayaswal, V. Virtual Mouse Control Using Hand Class Gesture.

[8] Abhilash, S. S., Thomas, L., Wilson, N., &Chaithanya, C. (2018). Virtual Mouse Using Hand Gesture. International Research Journal of Engineering and Technology (IRJET), 5(4), 3903-3906.

[9] Shibly, K. H., Dey, S. K., Islam, M. A., &Showrav, S. I. (2019, May). Design and development of hand gesture based virtual mouse. In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT) (pp. 1-5). IEEE.

[10]Nandwana, B., Tazi, S., Trivedi, S., Kumar, D., &Vipparthi, S. K. (2017, November). A survey paper on hand gesture recognition. In 2017 7th International Conference on Communication Systems and Network Technologies (CSNT) (pp. 147-152). IEEE.

[11] Ashok, M. A., Revathi, M. P., & Nirmal, B. (2020). FINGER RECONGITION AND GESTURE BASED VIRTUAL KEYBOARD. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 11(3), 2237-2245.

[12] Nikhil, C. D. S., Rao, C. U. S., Brumancia, E., Indira, K., Anandhi, T., &Ajitha, P. (2020, June). Finger Recognition and Gesture based Virtual Keyboard. In 2020 5th International Conference on Communication and Electronics Systems (ICCES) (pp. 1321-1324). IEEE.

[13] Shinde, O., Navale, K., Kunjir, D., More, A., &Taksal, A. GESTURE RECOGNITION BASED VIRTUAL MOUSE AND KEYBOARD.

[14] Gupta, R. G., Reddy, R. S., & Parmar, N. N. (2016). Virtual Keyboard Using Image Processing. International Research Journal of Engineering and Technology, 3(4), 2009-2013.

[15] Rustagi, D., Maindola, G., Jain, H., Gakhar, B., &Chugh, G. (2022). Virtual Control Using Hand–Tracking.
[16] Du, H., &Charbon, E. (2007, February). 3D hand model fitting for virtual keyboard system. In 2007 IEEE Workshop on Applications of Computer Vision (WACV'07) (pp. 31-31). IEEE.
[17] https://mediapipe.dev/
[18] https://www.autopy.org/
[19]https://www.w3schools.com/python/numpy/numpy_intro.asp