JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AI CONTROLLED GRAPHICAL BASED MOUSE POINTER USING HAND GESTURE RECOGNITION FOR EFFICIENT HUMAN COMPUTER INTERACTION

[1] Harini K, [2] Uma Maheswari. S

[1]Assitant Professor, [2]Professor,
[1] Electronics and Communication Engineering,
[1] Coimbatore Institute of Technology, Coimbatore, India

Abstract: The evolution of Human-Computer Interaction (HCI) has been from punch cards to keyboards and mice, and it is now moving towards graphical-based user interfaces. With the aid of image processing and computer vision, this research intends to develop a graphical-based Artificial Intelligence (AI) controlled mouse pointer using hand gesture recognition. The goal of this work is to concentrate on the efficient development of a virtual HCI device using web cameras. The system could track and react based on simple, dynamic hand motions and movements. AI is used to follow the motion of the hand by placing anchor points all over the hand. To get beyond the limitations of optical and wireless mice, this could eliminate the use of mechanical devices and wires. In order to make the interaction more effective and dependable, this research intends to develop a vision-based system to control numerous mouse functions via hand gestures.

Index Terms **- Artificial Intelligence (AI), Human-Computer Interaction (HCI), MediaPipe, landmark points, hand localization.**

I. INTRODUCTION

A software tool called the hand gesture-based virtual mouse enables users to control a device without using a real mouse. This research presents a novel computer-based system that utilizes hand gestures and hand tip recognition to create a virtual mouse, enabling users to generate mouse actions. The proposed device aims to replace the conventional mouse by utilizing the webcam or built-in digital camera of a laptop to perform cursor operations. To develop a hand gesture-based digital mouse, multiple image processing methods are used with a computer's web camera.

In this study, a user's hand motions are employed as mouse inputs. A web camera is a set of cameras that indefinitely take photos, and most laptops now include them. Security software that uses facial recognition has also made use of webcams to take advantage of face detection. Vision-based CC, which replaces the requirement for a computer mouse and mouse pad, can be used to fully utilize the capabilities of a system camera. They can also be used in HCI applications such as motion controllers and sign language databases, which can greatly benefit from using a system camera A wireless mouse controls the operations of a system camera. In contrast to a wireless or Bluetooth mouse, which requires a mouse, a dongle, and a battery to function, the client will use a built-in camera or a web camera to control the computer mouse with hand movements in this project.

A real-time hand tracking technology is proposed for various applications that can predict the human hand skeleton using just one RGB camera. The pipeline comprises two models, including a hand landmark model that predicts the hand's skeleton and a palm detector that identifies the hand's bounding box. The technology is implemented using MediaPipe, a framework for creating cross-platform ML solutions. The model and pipeline design facilitate in-the-moment gesture detection, which can effectively operate the mouse.

II. LITERATURE SURVEY

As intelligent technology for human-PC interactions becomes increasingly necessary in our everyday lives, a variety of mice with different shapes and sizes are available. These range from basic office mice to high-performance gaming mice. However, as it is discussed in [1] by R. Annachhatre, there are some limitations to this hardware as it's not as environmentally friendly as it.
Sande et al. [2] suggested the present virtual mouse control system comprises generally mouse operations that control the hand gesture-based virtual mouse, including left-click, right-click, and scrap-down, among other things, using a hand gesture detection system. Although there are several hand recognition systems, they chose static hand recognition, which is merely the recognition

of the shape created by the hand and therefore the definition of action for each shape made, which is confined to a few defined actions and generates a lot of confusion. There are more and more alternatives to using a mouse as technology progresses.

In a study by M. Van den Bergh [3], a system was proposed that utilizes hand gestures to perform various mouse actions such as left and right-clicking, scrolling, and other interactions. The research introduces a hand gesture-based interface that allows users to control a computer mouse through 2D hand gestures, aiming to enhance efficiency and reliability. Color detection algorithms based on cameras are used to detect hand movements. This technique primarily focuses on the effective use of a web camera to create a virtual device. The sensing principle employed to change the pointer on a computer screen involves locating the centroid of each input image, as hand movement directly influences the movement of the centroid.

Y. Li et al. [4] proposed a neural network-based architecture for key point detection using K-means cluster algorithms based on object key point similarity (OKS). It outperforms previous methods in both accuracy and speed. A single network based on pose anchors that can output the 2D hand pose directly from an entire image. A gesture recognition model is designed based on the input signal from the three-axis accelerometer [5]. It takes advantage of the user's hand movement to control the mouse movement, but the operational distance is limited to 15m. In [6], the left and right-click scroll up-down functions of a mouse are implemented by folding the first and middle fingers of the hand, respectively, and developing So, comparing the length of the fingers in the image with those in the image gives an idea about the functionality performed by the hand gesture-based virtual mouse.

S. Hazra and A. Santra [7] proposed the use of a millimeter-wave radar system for robust gesture recognition. The authors argue that traditional vision-based systems for gesture recognition are limited by factors such as lighting conditions, occlusion, and background clutter, and propose the use of radar systems as an alternative. Horatiu-Stefan [8] proposes a gesture recognition model using OpenCV and image processing algorithms such as contour detection, thresholding, and feature extraction. The authors use a combination of color-based and shape-based features to distinguish between different hand gestures.

Thakur et al. [9] proposed a gesture recognition model based on colour detection from images. Here, the person controlling the computer wears a colour band in his hand and the system recognizes the index finger based on the color codes. The gestures are recognized by the orientation of the index finger with respect to the other fingers. This model showed good accuracy in both low and high-intensity light environments. The limitation of this model is that it can only interpret the images in 2D and cannot be used for real-time tracking. Similarly, K. H. Shibly [10] proposed the use of colored caps on fingers, using which the user could control some of the computer cursor functions with their hands. Even though the use of a webcam or built-in cam to track fingers and then recognize gestures with that proved to be effective, there were still physical color caps being used, which might not be very user-friendly.

J. Farooq and M. B. Ali [11] proposed a machine-learning model to track hand skeletons using 21 landmark points and to recognize hand gestures in a video. This real-time tracker reduces the complexity by reducing the image quality. This predicts the hand skeleton from an RGB camera for AR/VR applications.W. A. A. Praditasar et al. [12] proposes a system consisting of a hand tracking module, a posture recognition module, a gesture recognition module, and a virtual mouse control module. The hand tracking module uses a depth camera to track the hand in real-time and extract its features, such as position and orientation. The posture recognition module uses a distance-based classifier to recognize hand posture based on hand features. The gesture recognition module uses a finite state machine to recognize hand gestures based on the sequence of detected postures. The virtual mouse control module maps the recognized gestures to mouse operations such as left-click, right-click, and scroll. It achieves high accuracy in recognizing both postures and gestures.

R. M. Gurav and P. K. Kadbe [13] utilized hand gestures to operate the robot or for applications in the home and business. A variety of machine learning techniques, including neural networks, support vector machines, and adaptive boosting, constitute the foundation of hand gesture identification systems (AdaBoost). AdaBoost-based hand-pose detectors were trained using a condensed Haar-like feature set as one of these techniques in order to strengthen the detector. For more than four hand motions, the associated context-free grammar-based suggested technique demonstrated excellent real-time performance with remarkable accuracy and resilience.

III. PROPOSED MODEL

The suggested virtual mouse solution works incredibly well to get around problems in the real world, such as situations where there isn't enough room to use a physical mouse button and for individuals who might have problems with their hands and aren't necessarily able to use a physical computer mouse. The proposed mouse can be used to get around these issues because palm gestures and fingertip detection can be used to control the specific PC mouse features while using a webcam. In the COVID-19 situation, it is also dangerous to use a particular device by physically contacting it because this may result in the virus spreading by pressing the device. The virtual mouse system uses MediaPipe and OpenCV frameworks to detect and track the fingertips of the hands, recognize gestures, and perform actions accordingly. The proposed flow of the system is shown in Fig. 1.

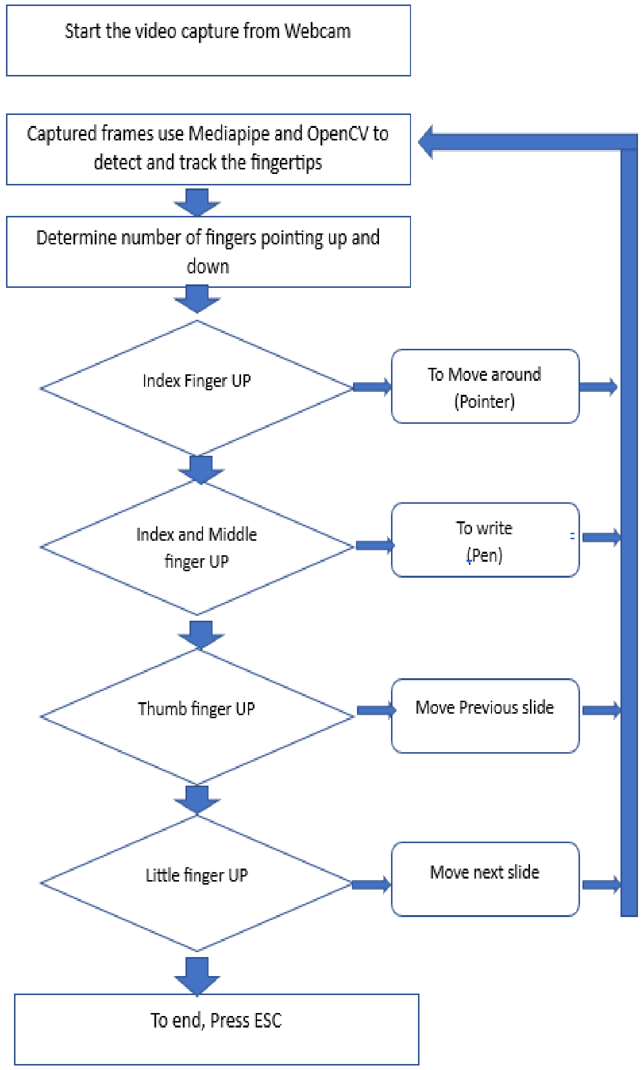

Fig. 1. Proposed flow of model

MediaPipe

MediaPipe is a framework that is employed for applying a machine learning pipeline and is associated with a Google open-source framework. Because the framework is built using statistical knowledge, the MediaPipe framework will help with cross-platform programming. The MediaPipe framework is multimodal, which means it can handle a wide range of audio and video formats. The MediaPipe framework is employed by the developers to create and analyze systems using graphs, as well as to create systems for application development. The steps involved in a MediaPipe-based system are specified as unit methods in the pipeline configuration. The pipeline will be able to run on a variety of platforms, allowing for quantifiability on mobile and desktop devices.

For several years, MediaPipe Box Tracking has facilitated real-time tracking in various applications such as Motion Stills, YouTube's privacy blur, and Google Lens, leveraging conventional computer vision techniques. The box tracking system takes image frames from a video or camera stream and computes the monitored box positions for each frame by starting box positions with timestamps, signaling 2D regions of interest to track. The starting box positions in this particular use case are determined by object detection, but the starting position can also be supplied directly by the user or by another system. Three key parts make up our solution: a motion analysis part, a flow packager part, and a box tracking part.

MediaPipe Box tracking and ML inference can be combined to create useful and effective pipelines. For instance, a pipeline for object detection and tracking can be built by combining box tracking with ML-based object detection. In comparison to executing detection on every frame (like MediaPipe Object Detection), this pipeline has the following benefits with tracking. It offers instance-based tracking, meaning that the object ID is preserved throughout multiple frames. Detection doesn't need to run every frame. As a result, it is possible to run more accurate detection models with larger detection models that are nevertheless lightweight and real-time on mobile devices. With the use of tracking, object localization is temporally consistent, resulting in less jitter being visible across frames.

OpenCV

To analyze and apply codes depending on a patient's policy coverage, the system may also read and assess an 834 form. One of the largest data dictionaries in the business and 50 million adjudicated claims were used to train the autocoder at first. MVP-1 had 89% accurate coding when it was delivered.

The system will become smarter as the product progresses through its MVP stages because it has been taught with 4 to 6 billion records and is anticipated to reach 98% coding accuracy. W. A. A. Praditasari [14] proposed a combination of hardware and software, including a webcam, an Arduino board, and OpenCV for hand tracking. The authors discuss the design of the virtual exhibits and the user interface, which allows visitors to navigate through the museum and interact with the exhibits using hand gestures. Varun et al. [15] used Numpy and OpenCV to create a pointer that will follow the rectangle based on the highlighted color. The color detection model will be able to detect one particular color in a colored picture.

IV. RESULTS AND DISCUSSION

The palm detection model was built and run successfully, with bounding boxes created every time the user's palm was in front of the camera. The landmark points (0 to 20) are localized, and gestures are developed. There are a total of four gestures in the model, each one signifying a certain cause.

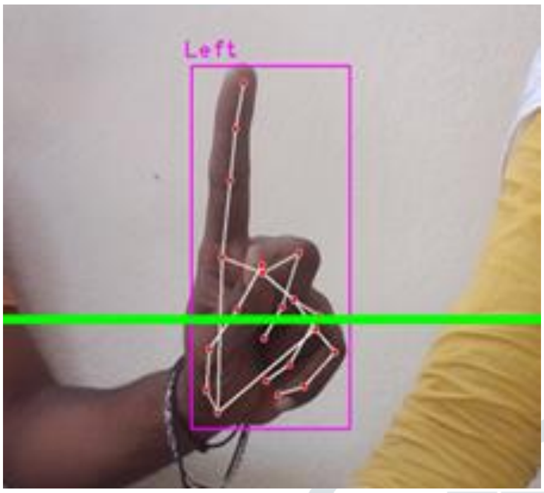

Fig. 2. (a) Pointer gesture

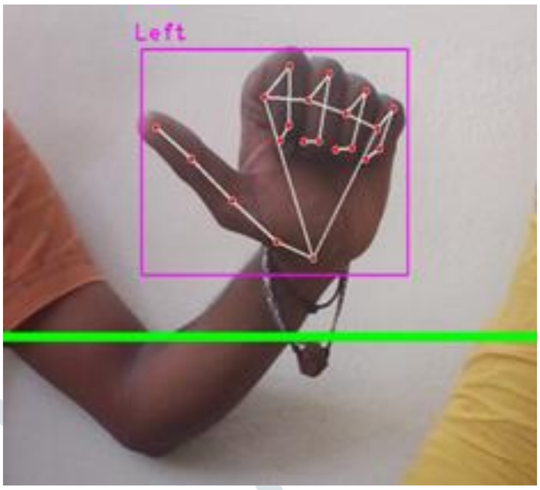

Fig. 2. (b) Left gesture

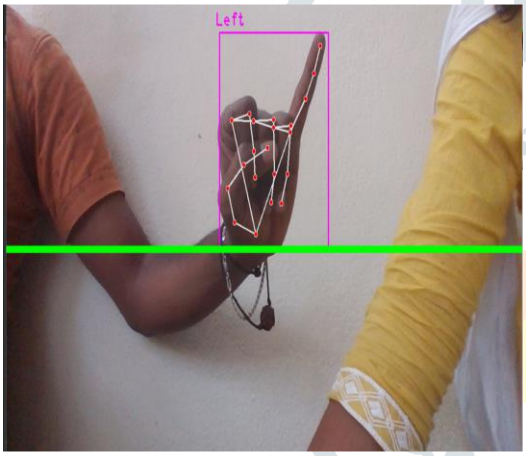

Fig. 2. (c) Right gesture

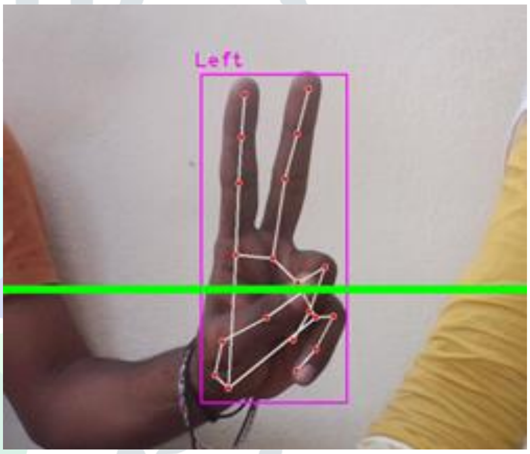

Fig. 2. (d) Virtual pen gesture

Pointer Gesture: The gesture shown in Fig. 2 (a) makes the user's index finger act like the mouse pointer. This gesture is not threshold-limited, as it works all across the screen.

Left Gesture: The gesture displayed in Fig. 2(b) enables the screen to navigate to the previous slide, such as taking the user to the final slide of PowerPoint.

Right Gesture: Fig. 2 (c) shows the gesture used to allow the screen to navigate to the next slide, for example, to move to the subsequent slide in PowerPoint. The left and right gestures are threshold-controlled in order to avoid missed signals, i.e., the gesture is only recognized and the action following the gesture is performed if the palm is above the threshold line. The threshold line can be altered according to need and purpose.

The Virtual Pen Gesture: As illustrated in Fig. 2 (d), this gesture enables the user to utilize their hand as a virtual pen, allowing them to make brief notes during the presentation. Fig. 3 (a) & (b) shows the result of the model and how the model interacts with the system by annotating and slide change.

Based on the obtained results and simulations, the model is suitable for real-time gesture tracking and can also be applied to various other applications. Additionally, more gestures could be incorporated into the model. Compared to previous models, this approach reduces complexity and computation requirements.

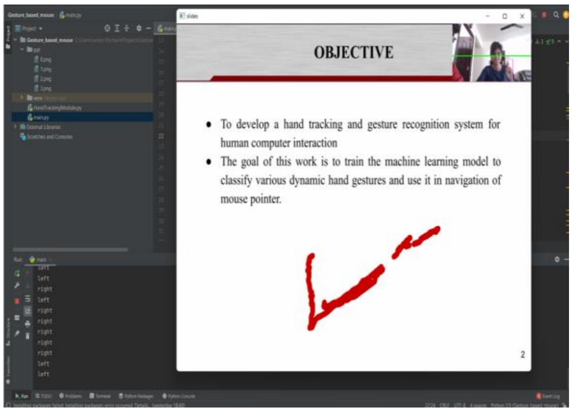

Fig. 6. (a) Annotations using virtual pen gesture

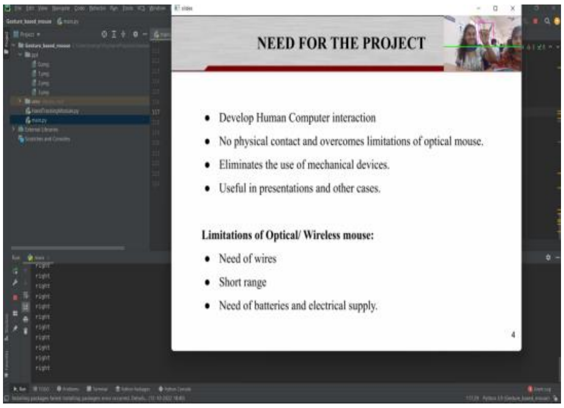

Fig. 6. (b) Slide change using left/right gesture

V. Conclusion and Future Scope

The AI visual mouse system is a revolutionary technology that uses artificial intelligence to enable users to control their computer with their eyes. This technology works by tracking the user's eye movements and translating them into commands. It is designed to be more intuitive and efficient than traditional mouse systems, allowing users to work faster and more accurately. With the help of AI, this system can also recognize objects in images and videos, making it easier for users to navigate through complex menus. The AI visual mouse system has numerous potential use cases in various industries, such as gaming, medical imaging, engineering design, and many more. The future scope of an AI visual mouse system is quite promising, with potential applications in various fields such as gaming, healthcare, and robotics. Here are some possible future developments for an AI visual mouse system:

Improved accuracy: As AI algorithms and computer vision technology continue to improve, the accuracy and precision of an AI visual mouse system can also be expected to improve. This could make it possible to use the system for even more complex tasks, such as controlling robotic arms or drones.

Enhanced accessibility: An AI visual mouse system could also have potential applications in healthcare and assistive technology for individuals with disabilities. For example, the system could control a computer cursor with eye movements or head movements, making it easier for individuals with limited mobility to use computers and other digital devices.

Integration with other AI technologies: An AI visual mouse system could be integrated with other AI technologies such as natural language processing and speech recognition to create a more seamless and intuitive user experience. This could allow users to control their devices using voice commands and other forms of input in addition to mouse movements.

Personalization: An AI visual mouse system could also be personalized to individual users, adapting to their unique movements and preferences over time. This could make the system more intuitive and efficient for each individual user.

REFERENCES

[1] R. Annachhatre, M. Tamakuwala, P. Shinde, A. Jain, and P. V. Kulkarni, "Virtual Mouse using Hand Gesture Recognition - A Systematic Literature Review," 2022 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), Pune, India, 2022, pp. 1-6, doi: 10.1109/ICBDS53701.2022.9935944.

[2] Riza Sande, Neha Marathe, Neha Bhegade, AkankshaLugade, Prof. S. S. Jogdand, "Virtual mouse using hand gesture," 2021 International Journal of Advanced Research in Science, Engineering and Technology (IJARSET) pp. 17124-17132, 2021.

[3] M. Van den Bergh and L. Van Gool, "Combining RGB and ToF cameras for real-time 3D hand gesture interaction," 2011 IEEE Workshop on Applications of Computer Vision (WACV), Kona, HI, USA, 2011, pp. 66-72, doi: 10.1109/WACV.2011.5711485.

[4] Y. Li, X. Wang, W. Liu and B. Feng, "Pose Anchor: A Single-Stage Hand Keypoint Detection Network," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 7, pp. 2104-2113, July 2020, doi: 10.1109/TCSVT.2019.2912620.

[5] M. S. Ghute, M. Anjum and K. P. Kamble, "Gesture Based Mouse Control," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 710-713, 2018, doi: 10.1109/ICECA.2018.8474905.

[6] V. V. Reddy, T. Dhyanchand, G. V. Krishna and S. Maheshwaram, "Virtual Mouse Control Using Colored Finger Tips and Hand Gesture Recognition," 2020 IEEE-HYDCON, Hyderabad, India, 2020, pp. 1-5, doi: 10.1109/HYDCON48903.2020.9242677.

[7] S. Hazra and A. Santra, "Robust Gesture Recognition using Millimetric-Wave Radar System," IEEE Sensors Letters, vol. 2, no. 4, pp. 1-4, Dec. 2018, Art no. 7001804, doi: 10.1109/LSENS.2018.2882642.

[8] Horatiu-Stefan Grif, Cornel Cristian Farcas, “Mouse Cursor Control System Based on Hand Gesture”, Procedia Technology, Volume 22, pp. 657-661, 2016, ISSN 2212-0173, doi:10.1016/j.protcy.2016.01.137.

[9] 1.S. Thakur, R. Mehra and B. Prakash, "Vision based computer mouse control using hand gestures," 2015 International Conference on Soft Computing Techniques and Implementations (ICSCI), pp. 85-89, 2015, doi: 10.1109/ICSCTI.2015.7489570.

[10] K. H. Shibly, S. Kumar Dey, M. A. Islam and S. Iftekhar Showrav, "Design and Development of Hand Gesture Based Virtual Mouse," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, pp. 1-5, 2019, doi: 10.1109/ICASERT.2019.8934612.

[11] J. Farooq and M. B. Ali, "Real time hand gesture recognition for computer interaction," 2014 International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE), Islamabad, Pakistan, 2014, pp. 73-77, doi: 10.1109/iCREATE.2014.6828342.

[12] A. Aksaç, O. Öztürk and T. Özyer, "Real-time multi-objective hand posture/gesture recognition by using distance classifiers and finite state machine for virtual mouse operations," 2011 7th International Conference on Electrical and Electronics Engineering (ELECO), Bursa, Turkey, 2011, pp. II-457-II-461.
[13] R. M. Gurav and P. K. Kadbe, "Real time finger tracking and contour detection for gesture recognition using OpenCV," 2015 International Conference on Industrial Instrumentation and Control (ICIC), Pune, India, 2015, pp. 974-977, doi: 10.1109/IIC.2015.7150886.
[14] W. A. A. Praditasari, R. Aprilliyani and I. Kholis, "Design and Implementation of Interactive Virtual Museum based on Hand Tracking OpenCV in Indonesia," 2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Semarang, Indonesia, 2021, pp. 253-256, doi: 10.23919/EECSI53397.2021.9624265.
[15] K. S. Varun, I. Puneeth and T. P. Jacob, "Virtual Mouse Implementation using OpenCV," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 435-438, doi: 10.1109/ICOEI.2019.8862764.