IJARSCT

ISSN (Online) 2581-9429

IJARSCT
Impact Factor: 7.301

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 3, Issue 5, April 2023

Hand Gestures Recognition For Virtual Mouse and Keyboard :A Novel Human- Computer Interaction Approach

Mr. P. Manikanda Prabu[1], Arivazhagi R[2], Deepika G[3], Dharshini J[4]
Assistant Professor, Department of Computer Science Engineering[1]
Students, Department of Computer Science and Engineering[2,3,4]
Anjalai Ammal Mahalingam Engineering College, Thiruvarur, India

Abstract: *Today's computer vision technology is so good that a machine can recognize its owner by using a simple picture-processing programme.. People use this vision in many parts of daily life in the current technological age, including face recognition, color detection, autonomous vehicles, etc. The One of the marvels of Human-Computer Interaction (HCI) technology is the mouse. Since a wireless mouse or Bluetooth mouse still need a battery for power and a dongle to connect it to the PC, they are not entirely device-free at this time. This study suggests an interactive computer system that converts handmotions into a virtual keyboard and mouse using computer vision. To carry out the task, we can use aninternal camera or an external camera. To operate the mouse and keyboard, we can use a built-in camera or an external camera. The computer's camera will scan the image of various hand gestures made by a person, and in accordance with the motions, the mouse pointer will move and even carry out various functions using various gestures. Other motions can also be used to access the keyboard's features. It will work as a virtual mouse and keyboard in the absence of any wiring or other hardware. The project's webcam is the only piece of hardware, and Python is used for all development on the PyCharm platform. Modern machine learning and computer vision techniques are used in this research to recognize hand gestures, and they operate without the need for any additional hardware.*

Keywords: Machine Learning, MediaPipe, Mouse, OpenCV.

I. INTRODUCTION

Image processing, a division of signal processing, can correspond of an image or a videotape as input and output as an image or different parameters of it. Gesture recognition and shadowing is a kind of image processing process. In recent times, a number of gesture recognition ways have been proposed. Hand tracking has several operations including motion capture, human-computer interaction and human behavior analysis. Several types of detectors and detection gloves are used for hand motion detection and tracking. Instead of using more precious detectors simple webcams identify the gesture and track the motion. Virtual Mouse and Keyboard derived from Gestures of Hand makes human interaction with computer simple by only making use of Hand Gestures. The computer requires almostno direct contact. All of the input and output operations can be virtually controlled by using static as well as dynamic hand gestures. While using wireless devices such as a wireless mouse or keyboard, it requires a dongle to connect to the PC, and it also requires a battery to power the mouse as well as keyboard for operating. Here the user uses his/her built-in camera or a webcam and hand gestures to control the computer mouse and keyboard operations. It has been considered that Hand gestures recognition is an important development technology in Human Computer-Interactions (HCI). They provide the computers the ability to capture the hand gestures and execute the assigned commands without even touching devices such as a mouse or a keyboard physically.

1.1 Keyboard

The virtual keyboard is an on-screen representation of a standard keyboard that can be controlled using hand gestures. The virtual keyboard can be used as an alternative to a physical keyboard, particularly in situations where a physical

DOI: 10.48175/IJARSCT-9313

keyboard may not be available or convenient to use, such asin virtual reality or augmented reality environments. The virtual keyboard can be controlled usinghand gestures that are recognized in real-time by the system. The gestures can be customized to mapto specific keyboard actions, such as pressing a key or typing a combination of keys. This allows the user to control the virtual keyboard using intuitive hand movements, without the need for physical contact. The virtual keyboard can be displayed on a variety of devices, such as a computer screen or a virtual reality headset. The user can interact with the virtual keyboard using hand gestures, which are recognized using machine learning techniques such as MediaPipe and OpenCV. This approach allows for a more natural and intuitive form of human-computer interaction, as the user can control the keyboard using hand movements that mimic real-world typing gestures.

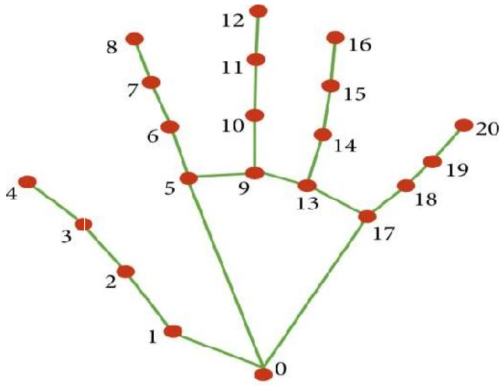
1.2 Machine Learning

Artificial intelligence (AI) systems may automatically learn from their experiences and get better over time this technique called machine learning. The creation of computer programmes that can access data and use it to learn for themselves is the focus of machine learning. To find patterns in data and improve future decisions based on the examples we provide, the learning process starts with observations or data, such as examples, firsthand experience, or instruction. The main goal is to let computers learn on their own, without help from humans, and alter their behavior accordingly. Algorithms for machine learning are frequently divided into supervised and unsupervised categories.In order to anticipate future events, supervised machine learning algorithms can use labeled examples to apply what they have learnt in the past to fresh data. The learning algorithm creates an inferred function from the examination of a known training dataset in order to forecast the output values. After sufficient training, the system is capable of providing objectives for any new input. The learning algorithm can also check its output against the desired, correct output to identify mistakes and fix the model as necessary. When the data used to train is neither classified nor are labeled, unsupervised machine learning algorithms applied. Unsupervised learning investigates how systems might extrapolate a function from unlabeled data to describe a hidden structure. Although the system is unable to determine the proper output, it explores the data and can infer hidden structures from unlabeled data using datasets. A learning method known as reinforcement machine learning algorithms interacts with the environment by taking actions and identifying rewards or errors. The most important aspects of reinforcement learning are trial-and-error searching and delayed rewards. With the use of this technique, machines and software agents may automatically select the best course of action in a certain situation in order to maximize performance. The reinforcement signal, or simple reward feedback, is necessary for the agent to learn which behavior is better.

1.3 MediaPipe

A framework called MediaPipe is a Google open source framework that is applied in a machine learning pipeline. Since the MediaPipe framework was created utilizing time series data, it can be usedfor cross-platform programming. The MediaPipe architecture supports multiple audio and video formats since it is multimodal. The MediaPipe framework is used by the developer to create and analyze systems using graphs as well as to create systems for application-related purposes. The pipeline configuration is where the actions in the MediaPipe-using system are carried out. Scalability on desktops and mobile devices is made possible by the pipeline's flexibility to execute on several platforms. The three essential components that make up the MediaPipe framework are performance evaluation, a system for accessing sensor data, and a group of reusable pieces known as calculators. A pipeline is a graph made up of units called calculators that are connected to one another by streams viawhich data packets pass. In order to create their own application, developers can add, remove, or redefine custom calculators anywhere in the graph. A data-flow diagram is produced by combining thecalculators and streams;

DOI: 10.48175/IJARSCT-9313



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGE
12. MIDDLE_FINGE
13. RING_FINGER_
14. RING_FINGER_
15. RING_FINGER_
16. RING_FINGER_
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Real-time detection and identification of a hand or palm is accomplished using a single-shot detector model. The MediaPipe uses the single-shot detector model. Because it is simpler to learn palms, the hand detection module initially trains a model for palm detection. Also, for small objects like hands or fists, the nonmaximum suppression performs noticeably better. Finding joint or knuckle coordinates inthe hand region is a model for a hand landmark.

1.4 Mouse

The virtual mouse is an on-screen representation of a standard computer mouse that can be controlled using hand gestures. The virtual mouse can be used as an alternative to a physical mouse, particularly in situations where a physical mouse may not be available or convenient to use, such as in virtual reality or augmented reality environments. The virtual mouse can be controlled using hand gestures that are recognized in real-time by the system. The gestures can be customized to map to specific mouse actions, such as moving the mouse pointer, clicking a button, or scrolling. This allows the user to control the virtual mouse using intuitive hand movements, without the need for physical contact. The virtual mouse can be displayed on a variety of devices, such as a computer screen or a virtual reality headset. The user can interact with the virtual mouse using hand gestures, which are recognizedusing machine learning techniques such as MediaPipe and OpenCV. This approach allows for a more natural and intuitive form of human-computer interaction, as the user can control the mouse using hand movements that mimic real-world mouse movements.

1.5 OpenCV

A computer vision package called OpenCV includes image-processing tools for object detection. Real-time computer vision applications can be created by utilising the OpenCV library for the Python programming language. The processing of images and videos as well as analytical techniques like faceand object detection rely on the OpenCV library.

II. RELATED WORK

2.1 Visual Panel

To solve the aforementioned issues, Zhengyou et al. presented the Visual Panel user interface system, which uses arbitrary quadrangle-shaped planar objects as panels and enables users to interact with computers using any tip-pointer tools. The movements of the tip-pointer will be recorded, analysed, and implemented, resulting in precise and reliable computer interface. In order to imitate both a mouse and a keyboard, the system as a whole comprises of a panel tracker, tip-pointer tracker, holography, computation and update, as well as action detector and event generator. Although the proposed systemaddressed the issue of cable length restrictions, it still needs a certain amount of space and materials to function. According to Zhengyou et al. , the system can accommodate any panel as long as it is quadrangle-shaped; hence, panels of any other form are not permitted.

2.2 Mouse Simulation Using Two Coloured Tapes

The issue at hand requires a ubiquitous computing solution, claim Kamran Niyazi et al. It was advised to emulate a color-tracking mouse as a result. Using computer vision technology, the aforementioned device tracks two coloured tapes on the user's fingertips. The pointer will be moved using one of the tapes, while mouse click events will be triggered by the other tape. The system must first distinguish the hand pixels from the rest of the image's pixels. This can be achieved by utilising a backdrop removal approach to decouple the movement information of the hands from the static background of the scene. The system must take two pictures of the static office from the viewpoint of the camera in order to accomplish this. The system will then carry out a second operation that divides the RGB pixels to compute the likelihood and separates the RGB values to identify the skin-covered and uncovered portions of the image. Once the procedure of subtraction is finished, this process will be finished. After the operation is finished, it will begin recognizing the desired colour in the image. To remove variations in hues of identical colours, the RGB image pixels will be converted into an HSV colour plane. The created image will go through noise-reduction filtering before being turned into a binary image. Even though the proposed method handled the most of the problems, it can only be usedto do simple cursor movements, left- and right-clicks, and double clicks. Further functionalitiesbecome available when the mouse scroll and centre click were ignored.

2.3 Virtual Mouse Using a Webcam

Kazim Sekeroglu also presented a device that uses three fingers and three different coloured points to imitate click actions. The suggested system can track the motion of the pointers, detect them using the defined colour information, move the cursor in accordance with the position of the pointer, and simulate both single and double mouse click events on the left or right. To detect the colours, they have leveraged the MATLAB's built in "imsubtract" function, with the combination of the noise filtering by utilising median filter, which are efficient in filtering out or at least minimize the "salt and pepper" noise. By separating the potential values for each pixel using MATLAB's built-in "im2bw" function, the collected picture will be converted to a binary scale image. After conversion, the captured image goes through another filtering step using "bwareaopen" to weed out the smallest areas in order to accurately count the number of objects that were detected in the image.

2.4 Portable Vision-Based Human Computer Interaction

Chu-Feng Lien also suggested "Ubiquitous Computing," which just needs fingertip input to control mouse cursor and click actions. The proposed system uses a feature called Motion History Images(MHI), a method that used to recognize movements with a row of photographs in time, instead of hand gestures or colour tracking to interact with the system. The proposed system does have its ownlimits, despite having good accuracy in a well-controlled setting. As a result of the frame rates being unable to keep up, the suggested system is unable to detect movements that move quickly, which increases the error rate. Also, because mouse click events only occur when the finger is in particular positions, this may require the user to move their fingers continuously in order to eliminate false alarms, which may be inconvenient

III. SYSTEM MODEL

3.1 Camera Module

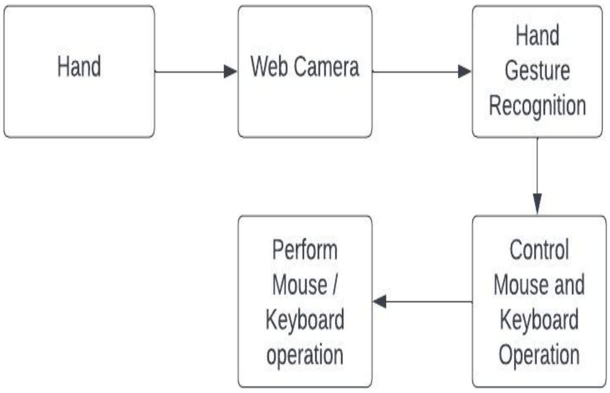
The camera module is an essential part of any hand gesture recognition system, as it captures the visual input needed for gesture recognition. The system works on the frames captured by the webcam on the computer machine or built-in camera on a laptop. By creating the video capture object, the system will capture video using a webcam in real-time. This camera will capture frame by frame and will pass it to the system.

3.2 Hand Detection Module

This module uses the MediaPipe Hand Detector API to detect and track hand landmarks, which can be used to determine hand gestures and movements. This module may involve configuring the MediaPipeAPI and processing the output to extract hand landmarks for further processing.

3.3 Mouse Module

The mouse module handles the mouse functionality, such as moving the cursor, clicking and scrolling.It receives input from the camera and hand detection modules which provides information about the position and movement of the user's hand and translates hand gestures into mouse actions. This module may involve implementing functions from Python libraries.


Hand
Web Camera
Hand Gesture Recognition
Control Mouse and Keyboard Operation
Perform Mouse / Keyboard operation

3.4 Keyboard Module

The keyboard module for hand gesture recognition in virtual mouse and keyboard technology is responsible for translating hand gestures into keyboard input, enabling the user to control the computer keyboard using hand gestures. The keyboard module typically receives input from the hand gesture recognition module, which provides information about the position and movement of the user's hand. Based on this information, the keyboard module can generate corresponding keyboard input. For example, making a fist with the hand can be recognized as a "space" keypress, and this can be used to insert a space character into the text being typed.

3.5 User Interface Module

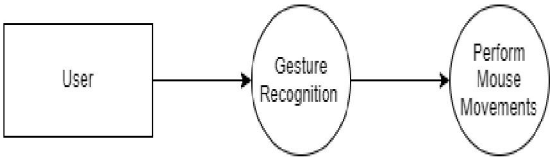
The user interface module for hand gesture recognition in virtual mouse and keyboard technology is responsible for providing a user-friendly interface that allows users to interact with the system and customize its behavior. This module provides a graphical user interface that displays the virtual mouse and keyboard to the user. It allows the user to select the input device,adjust settings.

IV. IMPLEMENTATION

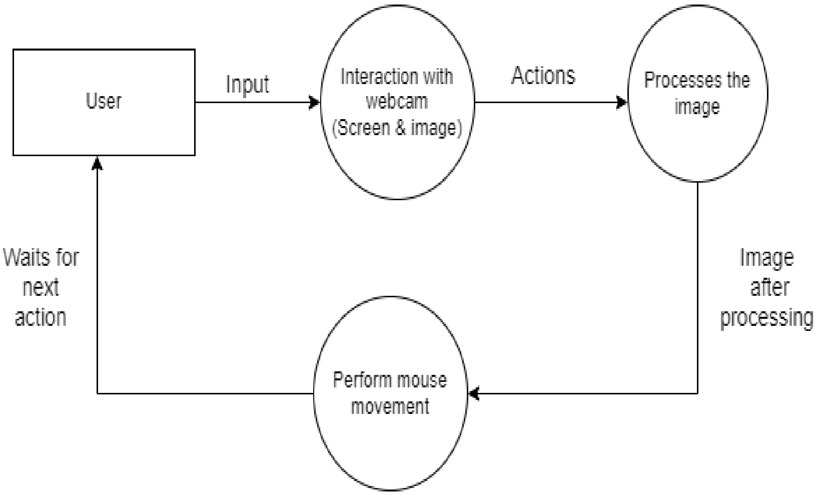
Algorithm

Step 1: Initialize the camera and set up the video stream.
Step 2: Load the hand detection module and set the detection parameters.
Step 3: Continuously capture frames from the video stream.
Step 4: Use the hand detection module to detect the presence of a hand in the frame.
Step 5: If a hand is detected, use the hand landmarks provided by the detection module to recognizethe hand gestures.
Step 6: Map the recognized hand gestures to the corresponding mouse or keyboard actions.
Step 7: Perform the mouse or keyboard actions accordingly.
Step 8: Display the virtual mouse and keyboard interface for the user to see the actions beingperformed.
Step 9: Repeat steps 3-9 until the user exits the program.

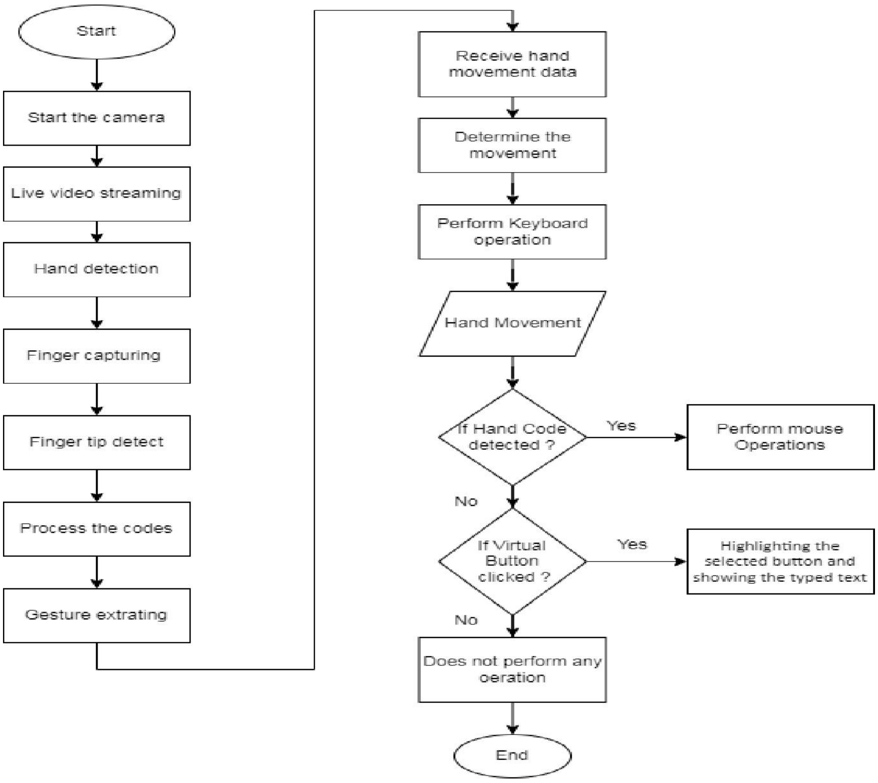
Data Flow DiagramLevel 0



Level 1



Flowchart



V. CONCLUSION

This idea envisions a system that would replace the keyboard and mouse and be able to recognize hand movements. This covers keyboard drag-and-click operations, mouse cursor movement, and additional keyboard features like printing alphabets. We have tried to implement AI Virtual mouse and keyboard using a real-time camera, hand detection, and tracking. It has several functions, such as right and double left clicks. The mouse's functionality is modelled like that of a real mouse and is based on computer vision methods. The alphabets are correctly aligned and the keyboard simulates a real keyboard. The outcome consists of key presses as well as space bar and backspace key functionality. The system was made to function under all circumstances. A bright room, hands, and an HD webcam are all that are needed for the system to function in all environments and produce superior results quickly. The effectiveness of the system will enable us to reduce the amount of space we need at work and prevent expenses from dust-related damage to physical keyboards and mice. To control programmes like games, media players, Windows Picture Manager, and power point presentations, however, we can use this approach.

REFERENCES

[1]. Ue-Hwan Kim, Sahng-Min Yoo, and Jong-Hwan Kim, “I-Keyboard: Fully Imaginary Keyboard on Touch Devices Empowered by Deep Neural Decoder”, IEEE Transactions on Cybernetics (Volume: 51, Issue: 9, September 2021).

[2]. Applying Hand Gesture Recognition for User Guide Application Using MediaPipe Indriani et.al. 2021.

[3]. Grishchenko I, Bazarevsky V, MediaPipe HolositicSimultaneous Face, Hand and Pose Prediction on Device,Google Research, USA, 2020, Access 2021.

[4]. J.Katona, “A review of human–computer interaction and virtual reality research fields in cognitive Info Communications,” Applied Sciences, vol. 11, no. 6, p. 2646, 2021.

[5]. An Interactive Computer System with Gesture-Based Mouse and Keyboard, by Dipankar Gupta et. al. 2020.

[6]. Vantukala VishnuTeja Reddy1, Thumma Dhyanchand2, Galla Vamsi Krishna3.“VirtualMouse Control Using Colored Finger Tips and Hand Gesture Recognition” (2020 IEEE HYDCON).

[7]. Jing-Hao Sun,Ting-Ting Ji,Shu-Bin Zhang “Research on the Hand Gesture Recognition Based on Deep Learning” 2020 12th International Symposium on Antennas,Propagation and EM Theory (ISAPE)

[8]. Kadir Akdeniz1 , Zehra C¸ ataltepe “Dynamic and Personalized Keyboard for Eye Tracker Typing” 2020 24th Signal Processing and Communication Application Conference (SIU).

[9]. Rafael Augusto Da Silva and Antonio Cl “Algorithm for decoding visual gestures for an assistive virtual keyboard” IEEE Latin America Transactions (Volume: 18, Issue: 11, November 2020).

[10]. Sugnik Roy Chowdhury Sathyabama, Sumit Pathak Sathyabama, M.D. Anto Praveena Sathyabama, “Gesture Recognition Based Virtual Mouse and Keyboard”, Proceedings of the Fourth International Conference on Trends in Electronics and Informatics (ICOEI 2020).

[11]. Glove-based hand gesture recognition sign language translator using capacitive touch sensor; K. S. Abhishek; L. C. F. Qubeley; D.Ho; 2018.

[12]. Using AEPI method for hand gesture recognition in varying background and blurred images; A.V.Dehankar; Sanjeev Jain; V.M.Thakare; 2017.

[13]. MotionFused Frames: Data Level Fusion Strateegy for Hand Gesture Recogniton; O. Kopukklu; N.Kose; G.Rigoll; 2018.

[14]. Real-time hand gesture recognition with EMG using machine learning, Andrés G. Jaramillo ; Marco E. Benalcázar, 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM).

[15]. Comparative study for vision based and data based hand gesture recognition technique, Oinam Robita Chanu; Anushree Pillai; Spandan Sinha; Piyanka Das, 2017 International Conference on Intelligent Communication and Computational Techniques (ICCT) .

[16]. Performance analysis of RTEPI method for real time hand gesture recognition, A.V. Dehankar; Sanjeev Jain; V. M. Thakare, 2017 International Conference on Intelligent Sustainable Systems (ICISS).

[17]. Using AEPI method for hand gesture recognition in varying background and blurred images, A.V. Dehankar; Sanjeev Jain ; V. M. Thakare, 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA).

[18]. Detection and recognition of hand gesture for wearable applications in IoMTW, Anna Yang ; Sung Moon Chun; Jae-Gon Kim, 2017 19th International Conference on Advanced Communication Technology (ICACT).

[19]. Hand Gesture Recognition System for Touch-Less Car Interface Using Multiclass Support Vector Machine, Mrinalini Pramod Tarvekar, 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS).

[20]. A New Dataset and Benchmark for Egocentric Hand Gesture Recognition; Y.Zhang; J.Cheng; H.Lu; 2018.

DOI: 10.48175/IJARSCT-9313