International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Virtual Mouse Using Mediapipe and OpenCV

Kushagra Pathak[1], Asst.Prof Akanksha Singh[2] , Asst.Prof Pradeep Kr Kushwaha[3]

[1] Dept. of Computer Science & Engineering Amity University India Kushagrapathak002@gmail.com

[2] Amity University India Asingh1@gn.amity.edu

[3] Amity University India pkkushwaha@gn.amity.edu

ABSTRACT –

The development of the PC mouse is one of the major advances in Human-Computer Interaction (HCI). Even while contactless or remote mouse try to minimize in-person interaction in today's tech-driven society, they still need batteries or other power sources. The necessity of reducing physical touch with shared gadgets has been emphasized even further by the COVID-19 epidemic. In order to overcome this difficulty, our virtual mouse system makes use of cutting- edge camera technology in order to identify finger and hand movements using PC vision. The system enables users to operate mouse tasks, including clicking, scrolling, and cursor movement, without the need of conventional input devices by utilizing the MediaPipe framework for gesture detection. By providing a touchless interface, adhering to current health recommendations, and increasing interaction efficiency, the virtual mouse technology improves user experience.

Index Terms: Media-pipe, HCI, real-time mouse system, deep learning foundation computer vision, Gesture Recognition

INTRODUCTION :

With the rapid advancement of technology, electronics are getting smaller every day. There are now wireless devices. This study suggests a mechanism that may cause some of the gadgets that will shape HCI (Human-Computer Interaction) in the future to become dormant. The idea is to employ gesture recognition to create a virtual mouse. The goal is to replace conventional or ordinary mouse devices with a simple camera to control mouse cursor functionalities. With just a camera, the Virtual Mouse functions as a conduit between the user and the computer. It facilitates mouse functions and allows the user to interface with a machine without the need for any physical or mechanical equipment. With a webcam or built-in camera holding a colored hat or colored sticky note paper, it is quite possible to record and track the fingertip of a hand using this gesture recognition technology. The system will then track the color and movement of the hand and move the cursor in tandem with it.

Typically we use a mouse, keyboard or other interacting devices which is mainly compact with the computer machine. The wireless devices also need a power source and connecting technologies, but in this paper, the user’s bare hand is the only input option using a webcam. So, it’s a very interactive way to control the mouse cursor.

This system is implemented in Python programming language using the Computer Vision based library OpenCV. This system
has the potential to replace the typical mouse and also the remote controller of machines. The only barrier is the lighting condition. That’s why the system still can’t be enough to replace the traditional mouse as most of the computers are used in poor lighting conditions.

Problem Description & Overview

The camera must be positioned such that it can observe the user's hands in the proper locations in order to track fingers as a moveable object and use them for mouse actions. This can be applied to patients who lack limb control, to save space, and to other situations of a similar nature. Rather than being a physical mouse, it is a virtual one that functions by tracking mapped fingertips and capturing frames and coordinates from a webcam.

Significance in Real World Application

Video conferencing is very popular nowadays. Because of this, the majority of computer users utilize webcams on their devices, and most laptops come with built-in webcams. The suggested webcam-based solution may be able to partially do away with the requirement for a mouse. Using hand gestures to connect with a computer is an intriguing and useful method of human-computer interaction, or HCI. On this interest, some excellent study has been done. Sign language recognition is another major application for hand motion recognition technology.

Objective

The intent is to create and put into practice a substitute system for controlling a mouse cursor. Using a webcam and a color detection technique, hand gesture recognition is an alternate option. The ultimate goal of this research is to create a system that uses any computer's color detection technique to detect hand gestures and control the mouse cursor.

RELATED WORK ON THIS THEORY

There are various uses for hand gesture-based cursor control applications, but most of them necessitate the usage of a DataGlove. The user-system performance efficiency is lowered as a result. Another problem with this technique is system complexity.

Two types of gesture recognition for HCI are possible: hardware-based and computer vision-based. Quam (1990) offered one of the first hardware-based systems, requiring the user to wear a cumbersome DataGlove in order to operate the system [1]. Even though this approach offers a high degree of control precision, it is quite challenging to use since some gestures are not appropriate for all users and are not practical for large numbers of people in daily life.

Marker-based and non-marker-based hand gesture recognition systems are also available for vision-based hand gesture recognition. In general, recognition that is not dependent on markers is less accurate than recognition that is. Furthermore, the accuracy of the maker-based recognition system is higher than that of other gesture recognition systems; nevertheless, the user must apply a basic color cap on their fingertip. However, this approach weighs nearly nothing in contrast to the hardware-based system's DataGlove. Gesture recognition may one day be used by computers to comprehend human body language. Instead of the rudimentary text-based interaction that currently exists, it will create a more robust interaction between humans and computers. The majority of marker-based gesture recognition mice track with at least two color markers. During performance, the system becomes sluggish and experiences some lagging as a result of detecting multiple colors. Based on an adaptive skin color model and a motion history image (MHI), ChenChiung Hsieh and Dung-Hua Liou wrote a paper titled "A Real-Time Hand Gesture Recognition System Using Motion History Image" [2] in 2010. They utilized a motion history image-based hand moving direction detection method and an adaptive skin color model in their work. The paper's primary limitation is a difficulty in working with more complex hand gesture recognition.

A human-machine interaction technique known as "A Human-Machine Interaction Technique: Hand Gesture Recognition Based on Hidden Markov Models with Trajectory of Hand Motion" [3] was written about by Chang-Yi Kao and Chin-Shyurng Fahn in 2011. Their work is very accurate, but it only worked on computers with a high configuration. Neethu, Angel, in 2013.

A paper titled "Real Time Static and Dynamic Hand Gesture Recognition" was proposed by P.S. [4] in which a practical framework for real-time human-computer interaction applications for gesture recognition is designed, developed, and studiedHowever, it could only be calculated in good light and could not function against a complicated background. 2013 saw Ashwini M. Sneha U. Patil The Monika B. Dudhane In a paper titled "Cursor Control System Using Hand Gesture Recognition" [5], Gandhi proposed the development of a machine-user interface that uses straightforward computer vision and multimedia techniques to perform hand gesture recognition. However, a significant impediment is prior to working with motion correlation calculations, skin pixel discovery and hand division from put away approaches should be finished.

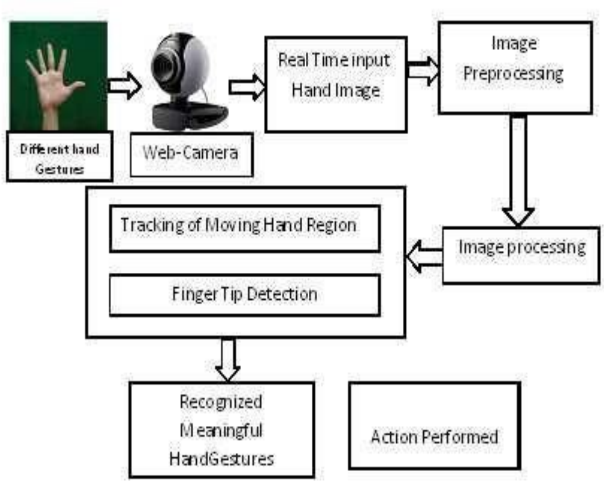
In a paper titled "Mouse Control using a Web Camera based on Color Detection" [6] published in 2014, Abhik Banerjee and Abhirup Ghosh used a color-detection-based camera to collect hand gestures. Their work is limited by the absence of brightly colored objects and a light operating background. It works well on some high-configuration computers. Based on directly extracting fingers from prominent hand edges, Yimin Zhou, Guolai Jiang, and Yaorong Lin published "A novel finger and hand pose estimation technique for real-time hand gesture recognition" [7] in 2016.

A paper titled "Virtual Mouse Using Hand Gesture" [10] was published in 2018 by Abhilash SS, Lisho Thomas, Naveen Wilson, and Chaithanya C. It was designed to work with the color detection system and works based on the number of colors that are detected. However, it cannot function without a static background and can only perform a few mouse actions. variety of modules and functions.

METHODOLOGY

We presented the block diagram and flow chart of Virtual Mouse Control Using Hand Gesture Recognition in this section. We also provided a brief description of the system's operation and components.

Fig-3.1 Block diagram for proposed system



A functional block diagram of the suggested system is shown in Fig. 3.1, which shows how the system functions. We have to raise our hand toward the webcam. The video is started and the frames are captured by the webcam. Pre- processing was applied to the supplied picture. Standardizing the picture is the fundamental purpose of the image pre-processor. Standardization is the process of resizing and pre-processing a picture to make its heights and widths comparable. The typical image is now enhanced utilizing an image processing approach in order to enhance its quality. After image processing, the camera moves its hand, and MediaPipe and openCV are used to identify the finger tips. It starts drawing after identifying the hand and tips of its fingers. The hand on the screen has a box surrounding it and hand landmarks. Draw a rectangle box
on the window PC to accommodate the mouse. It'll tell you which fingers are up and which are down. The mouse action is carried out based on the finger detections, and then the software goes back to the frames to carry out the subsequent action. The system functions in this manner overall.

ALGORITHM

Step 1: Start
Step 2: Start the webcam video capture and initialize the system.
Step 3: Frame capture with a webcam.
Step 4: Using Media Pipe and OpenCV, detect hands and hand tips and draw hand landmarks and a box around the hand.
Step 5: Draw a rectangle around the computer window area where we'll be using the mouse.
Step 6: Determine which finger is raised.
Step 6.1: The gesture is neutral if all five fingers are up, and the next step is taken.
Step 6.2: The cursor moves to step 2 if both the middle and index fingers are raised.
Step 6.3: A double click is performed when both index and middle fingers are joined side by side, and step 2 is performed.
Step 6.4: If both index and middle fingers are down, perform a left click and proceed to step 2.
Step 6.5: If the middle finger is down and the index finger is up, the right click is performed and the process proceeds to step 2.
Step 6.6: Volume up and down are accomplished by joining the thumb and index fingers and moving them up and down.
Step 7: To exit, press the EXIT key.

DETECTION MECHANISM

Every mechanism utilized in every component of the system this article proposes is detailed in detail.

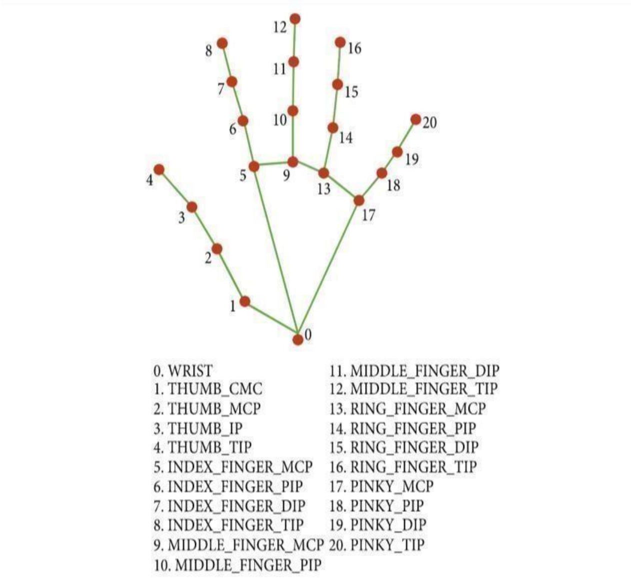
Camera

The system operates using frames captured by the computer's webcam or the built-in camera on a laptop. Utilizing OpenCV's cv2.VideoCapture, the video capture object is created to enable real-time video feed from the webcam. For accessing a single camera, the device index is set to "0". Additional cameras can be accessed by incrementing the index (1, 2, etc.). The camera captures each frame and provides it to the system for further processing.
Capturing The system employs the infinite loop to capture each frame from the webcam until program termination. The frames from the real-time video feed are processed, converting them from BGR to RGB color space for further analysis and movement detection for the mouse.

Hand Marker Detection

In this system, hand gestures are recognized using MediaPipe's hand tracking module. Key points on the hand, such as the fingertips and joints, are detected and used to determine the current gesture. The gesture recognition involves identifying the state of each finger based on its landmarks. Hand, such as the fingertips and joints, are detected and used to determine the current gesture. The gesture recognition involves identifying the state of each finger based on its landmarks

Fig-3.3 Model Graph of MediaPipe



- Identifying which finger is up and using the mouse. At this point, we identify which finger is up by utilizing the tip Id of the corresponding finger that was found using the MediaPipe and the corresponding coordinates of the up fingers, as seen in Fig. 3.3. We then execute the relevant mouse function.
- Mouse functions based on hand gestures and hand tip detection using computer vision.

GESTURE RECOGNITION

Mouse Movements

In the proposed system, mouse movements are controlled by recognizing specific hand gestures using MediaPipe's hand tracking capabilities. Each gesture
corresponds to a specific mouse function, such as cursor movement or scrolling. To facilitate smooth navigation, the index finger's coordinates are mapped from the camera frame to the screen coordinates, enabling the cursor to follow the user's finger movement in real time. MediaPipe's landmark detection provides the positional data, and smoothing algorithms ensure precise and responsive cursor behavior.
When the system detects two extended fingers (index and middle), it triggers upward scrolling by simulating mouse scroll events with pyautogui.scroll(). A fist gesture (all fingers curled) initiates downward scrolling. The detection relies on the relative y- coordinate movements of the detected hand landmarks; a decrease in y-coordinate values scrolls down, while an increase scrolls up.

Mouse Clicking

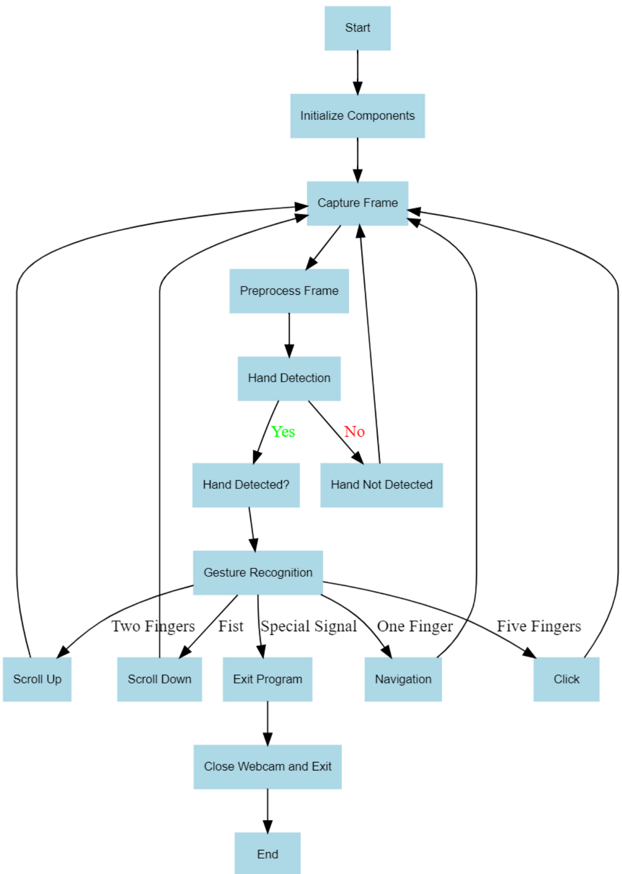
Mouse clicking is controlled by detecting the extension of all five fingers. When the system identifies all fingers extended, it simulates a mouse click using pyautogui.click(). This gesture mapping ensures that users can perform click actions with intuitive hand movements. The recognition of these gestures is facilitated by MediaPipe's accurate hand landmark detection, distinguishing between each finger's state.

Mouse Scrolling

In this approach, hand movements that are intuitive and that the Media Pipe framework detects are used to scroll. When two fingers (the index and middle) are extended, the system detects this and uses the pyautogui.scroll() method to smoothly scroll up. When a fist motion with all fingers curled is detected, the scroll-down action is initiated for downward scrolling. Real-time monitoring of hand landmarks, namely the analysis of the relative locations and motions of the fingers, is necessary for the scrolling capability.

The speed and direction of the scrolling precisely adapt to the user's hand motions because of the y-coordinate changes of the detected gestures are transferred to scrolling events. This method improves the entire user- computer connection by enabling a smooth, touch-free scrolling. The above method illustrates a methodical approach to imitating virtual mouse control with hand gestures.This will assist in converting hand gestures into mouse clicks. Figure 3.2's flow chart provides a diagrammatic depiction of the preceding procedure, or algorithm.

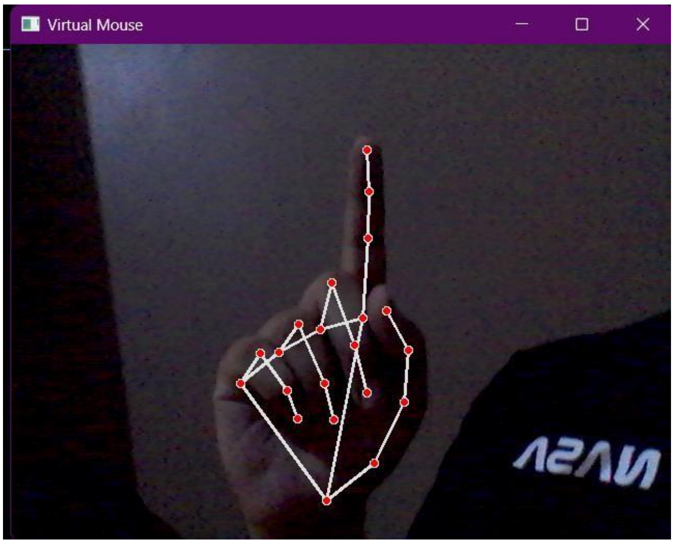
Figure 3.2 Flowchart for Proposed System



With the Virtual Mouse System, hand gestures may be used to control mouse functionalities via a camera. The technique is based on frames that are taken using a laptop or PC's camera. The camera records footage in real time using the OpenCV computer vision package for Python, as seen in Fig-3.1. Until the application ends, the virtual mouse system keeps processing the collected frames. Every frame that the camera records is sent to the system for additional processing.By Using recognized finger locations and configurations, this approach enables the use of hand movements to control mouse motions, clicks, and scrolling activities.

RESULTS & DISCUSSIONS

In order to improve vision-based human-machine interaction, we have used computer vision and human- computer interaction (HCI) methodologies in our study. This study suggests a method for utilizing hand motions captured by a camera to control mouse operations. The main features are the ability to move the mouse, click once or twice, and scroll using particular hand motions.Using a camera, the system records hand movements in real time, processing the gestures to carry out mouse activities. The system is able to recognize and understand commands like scrolling, clicking, and moving the cursor by recognizing the direction and location of fingertips. With its gesture-based interface, users can interact with computers in a natural and touch-free way, improving their experience across a range of settings without requiring extra hardware or depending on



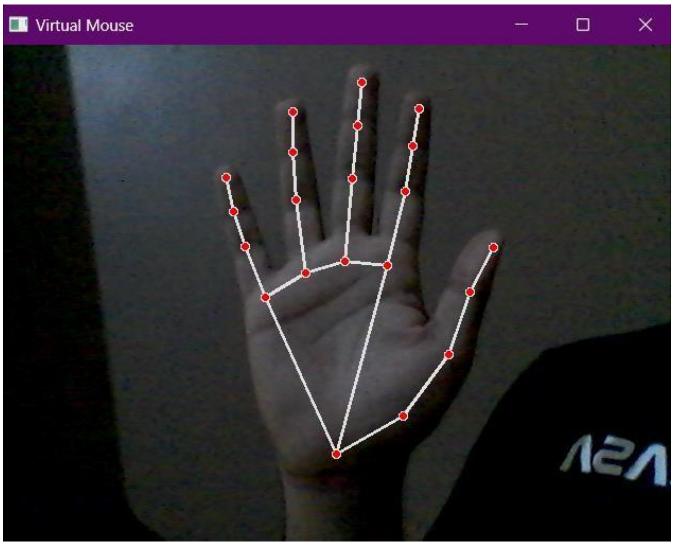


Figure A Neutral Gesture for Cursor Movement

Figure B Palm Gesture for Clicking

Using your index finger to stretch it will move the mouse. The system tracks the fingertip and converts its position into a movement of the pointer on the screen using MediaPipe and OpenCV. A smoothing algorithm makes sure that the pointer moves smoothly and precisely, which gives the user responsive and natural navigation

To click, extend all five fingers in the manner of a fully extended hand. This motion is recognized by the system as an instruction to click the mouse. Users may effortlessly engage with programs and interfaces since the system tracks the hand's position and makes sure all fingers are extended before sending a click input.

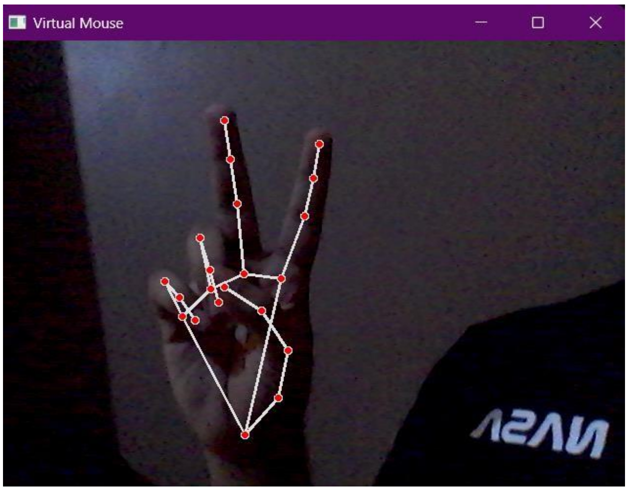


Figure C Index And Middle Finger for Scroll Up

By extending your index and middle fingers, you may scroll up. When the system recognizes this motion, an upward scroll is started. Because the identification is based on the relative locations of the fingertips, users may effortlessly scroll up pages or documents without using the mouse.

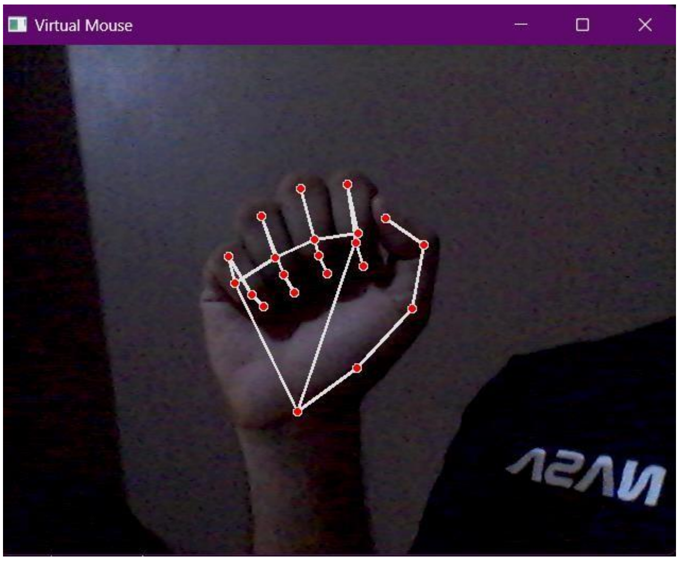


Figure D Fist Gesture for Scrolling Down

The user curls all of their fingers into a fist in order to scroll down. This gesture is recognized by the system, which then initiates a downward scroll. The virtual mouse technology executes the scroll motion by examining the landmark positions of the closed hand, providing a smooth and effective means of navigating through lengthy information.

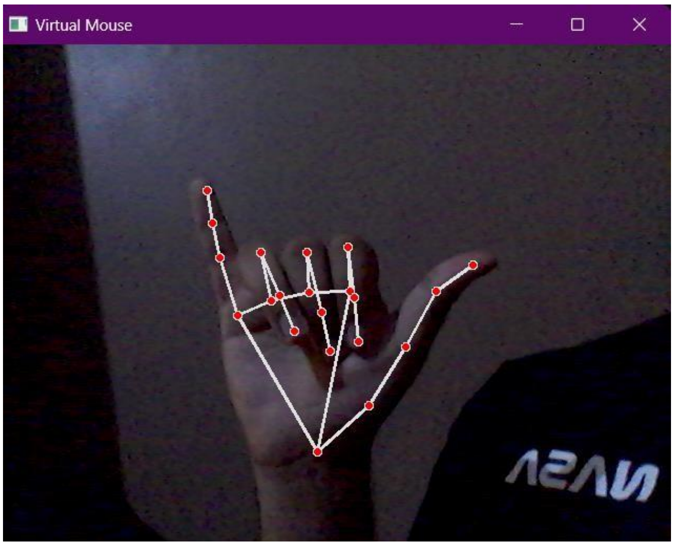


Figure E "Hang Loose" Gesture for Exiting thee loop and closing gesture recognition

Thumb and pinky extensions are used to do certain actions, such stopping an application, while curling the remaining fingers. This special gesture adds adaptability to the system by offering a natural approach to carry out commands that are different from standard mouse gesture operations.

EVALUATIONS AND GRAPHS

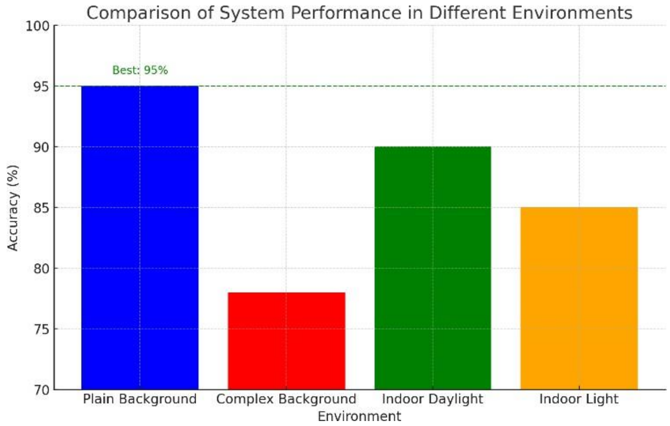
Testing was done for our project in a variety of settings to evaluate accuracy and performance:

a) *Simple Backgroun*d: When the system was first tested against a simple background, it showed quick reaction times and accuracy levels between 78% and 90%.

b) *Complex Backgrounds*: More difficult situations were tested using checkered shirts, colored T- shirts, and other common clothes worn while seated in front of a laptop or computer.

c) *Home & interior Environments*: A complicated home environment with typical interior illumination and day lighting was used to assess the system.

d) *Accuracy Evaluation*: To demonstrate the system's versatility and accuracy in a range of backdrops and lighting situations, the performance under these varied settings was measured, recorded, and shown in a table.



Graph 1 Comparison of System Performance In Different Environments

Parameters	Before	After
Plain Background	78	95
Complex Backgrounds	70	79
Home/Indoor Environments	65	87
Daylight Lighting	72	90
Artificial Light	78	85
Low Light	59	85
High Resolution Camera	76	95
Clear Gestures	78	87
Moderate Response	65	75
Ambiguous Gestures	59	75
Standard Resolution Camera	60	81

Table1. Table showing Comparison between Different Parameters

It is evident that the virtual mouse technology performs less well against busy and complicated backdrops. In spite of this, our technology operates faster and more responsively than previous gesture-based mouse control systems. The use of a high-definition camera can greatly increase the system's accuracy. The accuracy measurements shown in the table are based on experiments using a laptop's integrated VGA (Video Graphics Array) camera. Although Windows OS was used for system development and testing, Linux- or macOS- based systems may have a greater accuracy rate due to changes in operating system management and device integration.

APPLICATIONS

For a variety of use scenarios, the virtual mouse system provides a number of noteworthy benefits, including decreased hardware requirements and improved user engagement. Important uses consist of:

i. *Presentations*: Perfect for giving presentations without a real mouse or remote control, these allow for a simpler setup and less clutter on the workstation.

ii. *Robot Control*: Enables direct hand gesture control of robots, doing away with the need for extra hardware and offering a more user-friendly interface.

iii. *Digital art*: Increases creativity and workplace efficiency by enabling digital artists to produce 2D or 3D art on digital canvases with more flexibility and dynamic control.

iv. *Critical Events*: Practical in high-stakes settings where typical input devices may be unwieldy or unsuitable, such as battlegrounds, operating rooms, and mining fields.

v. *Virtual and Augmented Reality*: Provides a more pleasant and engaging gaming experience by enabling interaction without the need for conventional controls.

vi. *Sign Language Compatibility:* By converting hand gestures into computer commands, sign language communication helps the deaf and dumb communicate with one other and with computers more easily.

THE ACCURACY OVER TIME GRAPH

The Accuracy Improvement Over Time graph (Graph 2) in the assessment of our virtual mouse system shows a notable improvement in performance via iterative development. At first, the prototype phase's constraints limited the system's precision. On the other hand, we saw a significant improvement in accuracy after the first big update, which indicates that the important enhancements were successfully implemented.

The system was further improved by further upgrades, which resulted in gradual but steady improvements in accuracy. The system's final version shows an amazing accuracy rate, proving our development method was successful. This track of progress demonstrates how stable the virtual mouse technology is and how well it can function in real-world scenarios.

Graph 2 Accuracy over Time Graph

CONCLUSION :

A major breakthrough in human-computer interaction (HCI) has been made with the invention of the virtual mouse system, which combines gesture detection and computer vision technology. The possibility of using hand gestures to control mouse tasks has been effectively proven by this research, providing a fresh and simple way for users to engage with digital surroundings.

Our technology offers a flexible substitute for conventional input devices since it is based on real-time hand tracking and gesture detection. We have developed a user-friendly interface that transforms hand gestures into mouse commands by utilizing MediaPipe for hand recognition and OpenCV for real-time image processing. The system's primary features—mouse movement, clicking, scrolling, and unique gestures—are all intended to improve user control and engagement.

The flexibility of our system to adapt to different surroundings is one of its main advantages. In controlled environments, including simple backgrounds, the system performed admirably during testing, with accuracy ranging from 78% to 90%. The system's performance did, however, decline in complicated backdrops, highlighting the necessity for more calibration and improvement. Notwithstanding these difficulties, the system outperforms a lot of the current gesture-based interfaces in terms of speed and responsiveness, making it a viable substitute for people looking for a more natural way to connect.

To sum up, the virtual mouse technology is a big advancement in the area of computer-human interaction. Through the utilization of sophisticated computer vision algorithms and gesture identification, the system provides an adaptable, simple, and convenient substitute for conventional input modalities. Though there are obstacles to overcome, especially in intricate settings, the system's potential advantages for a range of applications and user groups make it a viable field for further research. To improve the accuracy and resilience of the system and enable wider implementation and integration into other digital contexts, more investigation and improvement will be necessary.

REFERENCES :

1. Chen-Chiung Hsieh ,Dung-Hua Liou and David Lee "A real time hand gesturerecognition system using motion history image" Proc.IEEE, 2010, pp. V2-394-V2-398 .
2. Kamran Niyazi, Vikram Kumar , Swapnil Mahe and Swapnil Vyawahare "Mouse Simulation Using Two Coloured Tapes",IJIST 2012, Vol.2, No.2, DOI : 10.5121.
3. Abhik Banerjee , Abhirup Ghosh , Koustuvmoni Bharadwaj , Hemanta Saikia "MouseControl using a Web Camera based on Colour Detection"IJCTT ,March 2014, volume 9 number 1, ISSN: 2231-2803
4. S. Thakur, R. Mehra and B. Prakash, "Vision based computer mouse control using hand gestures," International Conference on Soft Computing Techniques and Implementations (ICSCTI), 2015,

pp. 85-89.

5. Horatiu-stefan Grif, Train turc "Human hand gesture based system for mouse cursor control" INTER- ENG, October 2017.
6. Pooja Kumari ,Ghaziabad Saurabh Singh, Ghaziabad Vinay and Kr. Pasi "Cursor Control using Hand Gestures" International Journal of Computer Applications (0975 – 8887),2016.
7. Danling Lu, Yuanlong Yu, and Huaping Liu "Gesture Recognition Using Data Glove: An Extreme Learning Machine Method"Proc.IEEE,December 2016, pp. 1349-1354.
8. Alisha Pradhan , B.B.V.L. Deepak "Design of Intangible Interface for Mouseless Computer Handling using HandGestures" ICCCV(International Conference on Communication, Computing and Virtualization), 2016, oi: 10.1016/j.procs.2016.03.037.
9. Aashni Hariaa , Archanasri Subramaniana , Nivedhitha Asokkumara , Shristi Poddara and Jyothi S Nayak "Hand Gesture Recognition for Human Computer Interaction" Proc.ICACC(International Conference on Advances in Computing & Communications), 2017 August,pp367–374.
10. Abhilash S , Lisho Thomas, Naveen Wilson, and Chaithanya "VIRTUAL MOUSE USING HAND GESTURE" Proc. International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056 p-ISSN: 2395-0072, Apr 2018, Volume: 05 Issue: 04 .
11. S.Shriram, B.Nagaraj, J Jaya, S Shankar and P Ajay "Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to avoid Covid-19 Spread" Journal of Health Care Engineering, Volume:2021, Article ID:8133076.