STM JOURNALS

Current Trends in
Signal Processing

https://journals.stmjournals.com/ctsp

ISSN: 2277-6176 (Online)
ISSN: 2321-4252 (Print)
Volume 15, Issue 2, 2025
May–August
DOI (Journal): 10.37591/CTSP

Research CTSP

Generic Virtual Mouse

Pooja J. Pawar[1,*], Archana S. Banait[2]

Abstract

The mouse and keyboard have been replaced by new input mechanisms brought about by the quick development of Human-Computer Interaction (HCI). Using computer vision and deep learning techniques, this study explores the possibility of replacing actual mouse inputs in a computer system with hand gestures. We created a virtual mouse system that uses a normal webcam to record hand motions. Computer vision algorithms then process these gestures to handle mouse operations including clicking, scrolling, zooming, and moving the cursor. A range of situations, such as changes in hand size, gesture speed, and ambient lighting, were used to test the system. The system achieves 99% accuracy under ideal conditions, but it dips to 92% under dim or overexposed lighting and at distances longer than 30 cm, according to key findings. Moreover, it is still difficult to recognize intricate actions like scrolling and right-clicking. Future versions will include more training data for gesture refinement, adaptive illumination correction algorithms, and different recognition techniques to address these problems. The technology provides a creative and affordable solution for those with physical limitations in spite of these obstacles, and it holds out hope for future immersive hands-free digital system interaction.

Keywords: Computer vision, deep learning, hand gesture recognition, virtual mouse, human-computer interaction, python

INTRODUCTION

Conventional input devices, such the keyboard and mouse, have long been the main way to communicate with computers. Despite their efficiency, these gadgets have significant drawbacks, chief among them being their reliance on physical surfaces and limited mobility. For example, a mouse can only be used on a level surface, which restricts the user's range of motion. Additionally, both the mouse and keyboard can be uncomfortable or taxing to use for extended periods of time, especially for people with physical limitations [1]. These gadgets also depend on mechanical parts that need to be replaced and maintained on a regular basis because they are prone to wear and tear over time. Alternative approaches to engaging with digital systems that can get beyond these limitations and offer more adaptable, ergonomic, and accessible solutions are becoming more and more necessary as technology develops [2]. The usage of computer vision-based systems, which can completely do away with the requirement for physical input devices, is one promising answer to this issue.

In order to do tasks that a physical mouse would typically undertake, this study suggests a virtual mouse system that makes use of computer vision capabilities [3, 4]. Users can interact with the computer by just moving their hands in front of the camera thanks to the system's usage of a conventional webcam for hand gesture recognition and fingertip detection. By doing away with the necessity for a physical mouse or keyboard, this method improves accessibility and usability, especially for people with mobility disabilities.

***Author for Correspondence**
Pooja J. Pawar
E-mail: poojapawar7841@gmail.com

[1]Research Scholar, Department of Computer Engineering, Mumbai Education Trust, Bhujbal Knowledge City, Nashik, Maharashtra, India
[2]Assistant Professor, Department of Computer Engineering, Mumbai Education Trust, Bhujbal Knowledge City, Nashik, Maharashtra, India

Received Date: May 19, 2025
Accepted Date: May 24, 2025
Published Date: June 25, 2025

Citation: Pooja J. Pawar, Archana S. Banait. Generic Virtual Mouse. Current Trends in Signal Processing. 2025; 15(2): 26–32p.

The main goal of this research is to overcome typical issues such inaccurate gesture detection and the need on physical markers, which are frequently necessary by other gesture-based systems, in order to address the shortcomings of current systems. This system provides a more accurate and user-friendly way to operate the cursor and carry out other mouse tasks by detecting and interpreting hand gestures using machine learning techniques. Figure 1 displays the block diagram. In order to develop more inclusive and user-friendly interaction techniques for a broad spectrum of users, this study investigates how computer vision and hand gesture recognition can produce a more seamless, hands-free user experience.

LITERATURE REVIEW

Concerns about accessibility and user experience are addressed by research on combining voice commands with hand gestures for computer interface. Research looks into voice command productivity and gesture-based virtual world manipulation presents a method for controlling mouse movement that combines voice commands and hand motions. This strategy has the potential to improve computer inclusivity and user interaction. To solve technological issues and usability concerns for practical applications, more research is essential [1, 5].

By introducing a novel technology, the article by Shibly *et al.* on the "Design and Development of Hand Gesture Based Virtual Mouse" advances the rapidly developing field of human-computer interaction. The article, which was presented at ICASERT 2019, describes the development of a hand gesture-controlled virtual mouse that demonstrates improvements in user-friendly interface design. The need for alternate input modalities is addressed by the work of Shibly *et al.*, which is especially pertinent in situations where conventional devices might not be feasible or accessible. This study creates new possibilities for improving accessibility and user experiences in computing settings by proving that hand gesture-based interaction is feasible [2].

The promise to transform human-computer interaction has led to a surge in research on hand gesture recognition for virtual mouse events. Diverse approaches and algorithms are investigated in studies to accurately understand hand gestures in real-time. Ranawat *et al.* make a contribution to this area by introducing a system that uses hand gesture recognition to operate virtual mouse events. These developments open the door to paradigms of interaction that are more natural and intuitive, improving accessibility and user experience in computing settings [3].

In recent years, there has been a lot of interest in the idea of a virtual mouse that is controlled by hand movements. Using computer vision and machine learning techniques, researchers have investigated a number of methods for putting such systems into practice. By introducing a hand gesture-based virtual mouse system, adds to this corpus of work.

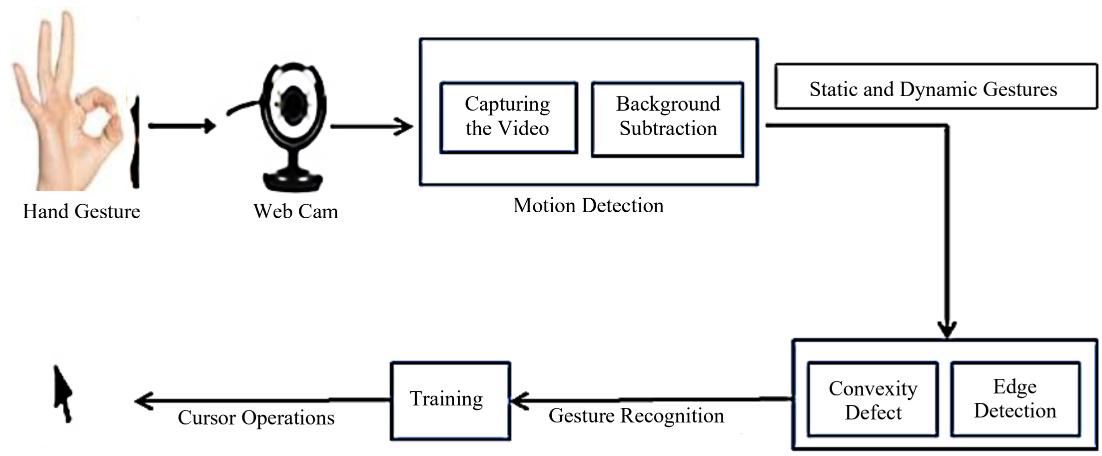


Figure 1. Block diagram of proposed systems.

This work demonstrates how gesture-based interaction can be incorporated into computing environments to give people a different and possibly more natural way to engage with digital interfaces [4]. A thorough analysis of hand gesture recognition techniques is given by the evaluated literature, which includes surveys, real-world applications, improved recognition techniques, application-specific research, and comparative analyses. Together, these works advance the state-of-the-art in hand gesture recognition and open the door to creative HCI solutions in augmented reality, virtual worlds, and other fields.

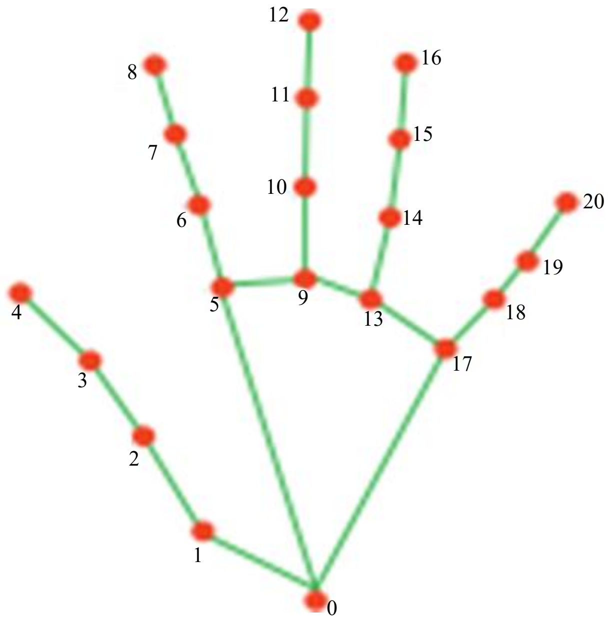
TECHNIQUES IMPLEMENTED FOR THE DEVELOPMENT OF MODEL

In order to carry out mouse operations based on hand movements, the generic virtual mouse system combines hardware, software, and machine learning approaches. The main elements and techniques utilized in the system's creation are listed below:

- *Hardware:* The system makes use of a typical webcam that is built into a desktop or laptop computer. The webcam records live footage of the user's hand movements, which are then processed to operate the mouse [6].
- *Software:* The Python programming language is the main foundation for the software implementation because of its adaptability and the abundance of strong libraries for machine learning and image processing. Among the important libraries utilized in the development are:
  - *OpenCV:* OpenCV (Open Source Computer Vision Library) is employed for image processing tasks, such as background subtraction, image filtering, and feature extraction [7].
  - *MediaPipe:* MediaPipe, a library developed by Google, is used for real-time hand tracking and recognition. It facilitates the identification of key hand landmarks and enables the accurate detection of hand gestures shown in Figure 2(a and b).

Hand Detection and Gesture Recognition

The system uses MediaPipe's hand-tracking features to identify 21 important hand landmarks, such as the palm and finger tips [8]. By examining the relative locations of these landmarks, the hand gestures are identified. Based on the hand placements and movements shown in Figure 3, the gestures are classified using machine learning techniques known as Convolutional Neural Networks (CNN). The real-time video feed is processed by the system to extract features, which are subsequently translated into particular mouse operations.



(a)

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_FINGER_DIP
12. MIDDLE FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY DIP
20. PINKY_TIP

(b)

Figure 2. (a, b) Identification of key landmarks.

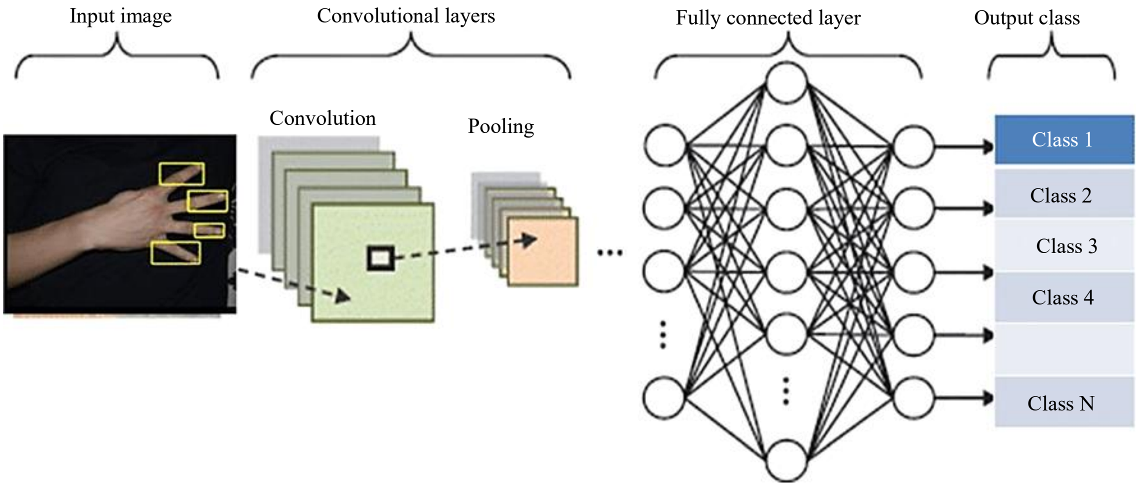


Figure 3. Architecture of CNN.

Motions and Corresponding Functions

The mouse can be controlled by the following hand motions on the system:

- *Movement of the Cursor:* The index finger can be moved in any direction. To move the pointer, the screen coordinates are transferred to the finger's position in relation to the webcam.
  - *Left Click:* The system recognizes a motion in which the thumb and index finger are brought together. This function is treated as a left-click.
  - *Right Click:* When the index and middle fingers are lifted and brought close together, the system detects it. The right-click action is assigned to this gesture.
  - *Scrolling:* The system detects finger movements that are vertical and interprets them as scroll actions, which shift the screen's content up or down.

Data Collection and Training

A wide range of hand gesture data, recorded in various lighting scenarios and hand sizes, are used to train the system in order to increase accuracy and recognition. To assist the model generalized among users, the training data include a range of motions made by various people [9]. Real-time processing is used to identify and categorize the hand motions. In order to identify the user's hand movements and translate them into the appropriate mouse commands, the system continuously analyses the video feed. The gestures are converted into click commands or cursor movements, which are then displayed on the screen.

Performance Evaluation

A range of situations, such as varying hand sizes, speeds, and ambient lighting, are used to test the system. Performance is assessed using the following criteria:

- *Accuracy:* The proportion of correctly recognized gestures.
- *Responsiveness:* How long it takes to react to gestures and take action.
- *Usability:* The system's ease of use and user pleasure. To create a user-friendly and precise virtual mouse system, the methodology integrates machine learning, hand gesture detection, and real-time picture processing. Because of its adaptability and simplicity of use, the system has the potential to improve accessibility and offer a hands-free substitute for conventional input devices [10].

THE VIRTUAL MOUSE SYSTEM'S WORKFLOW

The system's workflow includes a number of crucial processes that guarantee precise hand gesture recognition and the appropriate mouse action execution:

- *Recording Webcam Video Input:* Real-time video frames are continuously streamed by the webcam and processed by the system. OpenCV is used to process the frames and manage video input effectively.

- *Identifying Hand Motions with MediaPipe:* The hand tracking model from MediaPipe recognizes 21 hand landmarks. These landmarks are employed to monitor hand motions and identify particular gestures [7].
- The system can identify a variety of hand movements, like a pinch between the thumb and index finger or an elevated index finger, and classifies them according to the relative locations of the landmarks.

Connect Mouse Actions to Gestures

The system associates mouse actions with observed movements using PyAutoGUI. For instance:

- The movement of the cursor is controlled by an elevated index finger.
- A left click is triggered by a pinch gesture, which combines the thumb and index finger.
- Scrolling actions are matched to finger movements.
- The transformation of gesture data into commands that replicate mouse actions on the screen is managed by the Python libraries.

Carry Out Corresponding Measures

- PyAutoGUI performs the appropriate action, such as moving the cursor, clicking, or scrolling, after recognizing the gesture and mapping it to a mouse function.
- To guarantee responsive and seamless interaction, the system keeps an eye on the hand motions and updates the actions in real-time [3].

The System Assessment and Performance accuracy of gesture detection, responsiveness, and usability were among the factors used to assess the system's performance. To guarantee robustness in a range of illumination and hand sizes, testing was done in a variety of environmental settings.

FINDINGS AND DISCUSSION

The outcomes of the AI virtual mouse system's testing and assessment are shown in this section. The device replaces conventional mouse inputs with hand motions recorded by a typical webcam and processed using computer vision algorithms (Figure 4).

Conditions of Testing

The system was tested at different distances from the webcam and under different lighting conditions:

- Lighting Conditions: Hand landmark detection was impacted by low or overexposed lighting, where accuracy fell to 92%, however it was high under normal lighting.
- Distance Sensitivity: When the hand was within 15 cm of the webcam, performance was at its peak. Tracking irregularities were seen beyond 30 cm, which resulted in decreased precision for intricate motions like the scrolling and right-clicking depicted in Figure 4.

Future developments will concentrate on hardware advancements including infrared-based hand tracking and adaptive brightness adjustment using AI-based filtering approaches to lessen these problems.

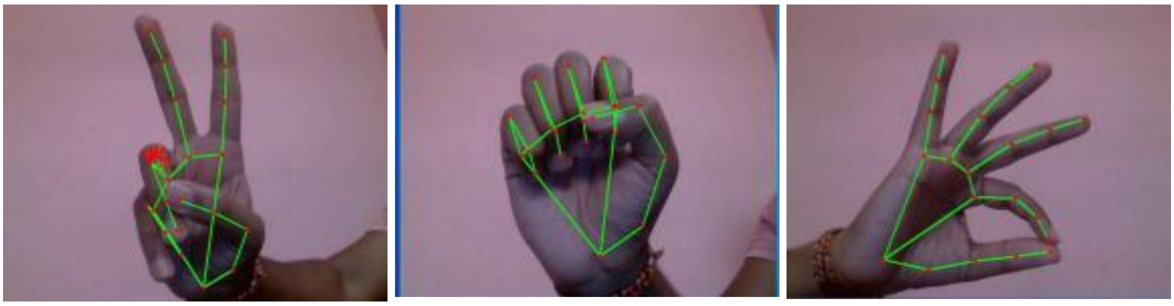


Figure 4. Various functionalities of generic virtual mouse.

Complex Gesture Challenges and Gesture Recognition

Complex motions like right-clicking and scrolling presented difficulties, whereas simple gestures like left-clicking and mouse movement worked well. Sometimes there were misunderstandings because the system had trouble identifying the difference between small finger movements.

Anticipated Improvements

- *Additional Training Data:* Adding more complicated gesture variations to the dataset.
- *Advanced Machine Learning Models:* To enhance sequential gesture detection, sophisticated neural networks, like CNN-LSTM hybrids, are being implemented.
- *Gesture Calibration Feature:* This feature lets users adjust the sensitivity of their gestures for better accuracy.

Evaluation via Comparison

Although the research offers a comparative analysis, direct comparisons with current systems were hampered by dataset restrictions. Future versions will benchmark performance against open-source datasets for hand gesture detection in order to improve evaluation [6]. Create standardized test cases so that the system may be compared to other implementations of virtual mice.

Important Results

The system's performance demonstrates how hand gesture recognition can be used in HCI. To increase usefulness, however, the issues of distance sensitivity, low-light performance, and right-click detection must be resolved. Future versions will include:

- Deep learning-based image improvement for adaptive lighting adjustment.
- Infrared-assisted gesture recognition for better low-light performance.
- CNN model optimization with more training data for more complicated gesture identification.
- Integration with more durable hand-tracking systems, like cameras that detect depth.

The system can offer a more dependable and convenient substitute for conventional input devices by resolving these issues.

CONCLUSION

As a hands-free substitute for conventional input devices, the AI virtual mouse technology shows great promise. Under ideal circumstances, great accuracy can be achieved, although handling environmental variables and complex gesture recognition still present difficulties. Future versions will further develop the system, making it more accurate, efficient, and adaptive for real-world applications by including improved datasets, hardware improvements, and sophisticated deep learning models.

REFERENCES

1. Palsodkar P, Pathak A, Khawle K, Srivastava K, Fulzele M, Khurge D. Controlling Mouse Movement Using Hand Gestures and Voice Commands. In 2023 IEEE 4th International Conference for Emerging Technology (INCET). 2023 May 26; 1–5.

2. Shibly KH, Dey SK, Islam MA, Showrav SI. Design and development of hand gesture based virtual mouse. In 2019 IEEE 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). 2019 May 3; 1–5.
3. Ranawat M, Rajadhyaksha M, Lakhani N, Shankarmani R. Hand gesture recognition based virtual mouse events. In 2021 IEEE 2nd International Conference for Emerging Technology (INCET). 2021 May 21; 1–4.
4. Matlani R, Dadlani R, Dumbre S, Mishra S, Tewari A. Virtual mouse using hand gestures. In 2021 IEEE international conference on technological advancements and innovations (ICTAI). 2021 Nov 10; 340–345.
5. Dahake RP, Kharat MU. Face recognition from video using threshold based clustering. Turk J Comput Math Educ. 2021; 12(1S): 272–85.
6. Zhou Q, Fitzmaurice G, Anderson F. In-depth mouse: Integrating desktop mouse into virtual reality. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. 2022 Apr 29; 1–17.
7. Volk T, Althoff F, Lang M, Grahn H. A Generic User Interface Framework. In Proc of the 9th Int Conf on Human-Computer Interaction. 2011 Aug 1; 1–9.
8. Pointecker F, Schwajda D, List D, Anthes C. A generic architecture for cross-virtuality. In: ISS'21 Workshop Proceedings: Transitional Interfaces in Mixed and Cross-Reality: A New Frontier?; 2021 Nov 14; Łódź, Poland. New York (NY): ACM; 2021. p. 1–6. Available from: https://doi.org/10.18148/kops/352-2-1l3l2wncu3tgb5.
9. Althoff F, Volk T, McGlaun G, Lang MK. A generic user interface framework for virtual reality applications. In Proceedings 9th HCI (HCI International 2001) Conference, New Orleans, Louisiana, USA. 2001; 5–10.
10. Salmeron-Majadas S, Baker RS, Santos OC, Boticario JG. A machine learning approach to leverage individual keyboard and mouse interaction behavior from multiple users in real-world learning scenarios. IEEE Access. 2018 Jul 20; 6: 39154–79.

