International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 3, 2025

Gesture Controlled Virtual Mouse

Sandyapagola Suchitha -Roll No. 24N81D5805

Dr. V. S. GiridharAkula ,Dr.M.Ramesh

Dept. of Computer Science and Engineering,

Sphoorthy Engineering College, Hyderabad, India.

Approved by AICTE, Affiliated to J.N.T.U, Hyderabad

ABSTRACT

Human–computer interaction has evolved significantly with advancements in artificial intelligence (AI) and computer vision. Traditional pointing devices such as the mouse and touchpad are widely used but present challenges in terms of accessibility, portability, and physical limitations. This paper presents a gesture-controlled virtual mouse system that leverages real-time hand gesture recognition to perform mouse operations without physical hardware. The proposed system integrates MediaPipe for landmark-based hand tracking, OpenCV for image processing, and Python libraries such as PyAutoGUI and Pynput for GUI automation. The approach eliminates additional hardware requirements and ensures cost-effectiveness. Experimental results demonstrate successful implementation of functions such as cursor movement, left and right clicks, drag-and-drop, multi-selection, and screenshot capture. The system improves accessibility, offering touchless interaction for users with physical disabilitiesand potential applications in healthcare, education, and hazardous environments.

Keywords— Virtual Mouse, Hand Gesture Recognition, Human–Computer Interaction, OpenCV, MediaPipe, Machine Learning

1.INTRODUCTION

Advances in computer vision and AI have transformed human–computer interaction (HCI). Conventional devices such as the mouse and keyboard are effective but have limitations for users with mobility impairments and in contexts where physical contact is inconvenient. Gesture recognition provides a natural and intuitive mode of interaction, enabling users to control systems through hand movements without physical devices.

This work introduces a gesture-controlled virtual mouse system implemented in Python using MediaPipe and OpenCV. By detecting hand landmarks in real time, the system maps gestures to standard mouse functions, offering a portable and cost-efficient solution to improve accessibility and usability.

2.LITERATURE SURVEY

Previous research in gesture recognition highlights the potential and challenges of contactless interfaces.

Dudhane et al. (2013) proposed a gesture-based cursor control system, but the need to store and process frames slowed performance.

Liou et al. (2010) developed a system using motion history images, but it struggled with complex gestures.

ISSN 2347–3657

Volume 13, Issue 3, 2025

Quam (1990) presented glove-based gesture recognition with high accuracy but limited usability due to restricted hand movement.

Singh et al. (2016) proposed a color-band approach for gesture recognition, requiring external markers, reducing practicality.

3.METHODOLOGY

The proposed system employs a webcam to capture real-time video frames, which are processed to recognize gestures. The workflow includes:

1. Input acquisition via webcam.
2. Hand landmark detection using MediaPipe.
3. Gesture recognition by analyzing landmark distances.
4. Mapping gestures to mouse actions.
5. Execution of operations using PyAutoGUI and Pynput,

This pipeline ensures smooth and accurate interaction without specialized hardware.

4.EXISTING SYSTEM

One popular example is **Camera Mouse**, which tracks head and handmovements to control the cursor. **GesturePak**and **Leap Motion** are other tools that allow gesture-based interaction with computers using sensors or webcams.**Media Pipe** by Google provides hand-tracking solutionsthatdevelopersusetocreatecustomgestur e control systems.Somesoftware requires users to weargloves or use colored markers for better accuracy. Others, like **HandTrack.js**, work directly in web browsers without extra hardware.

Using the current system, although there are a number of quick access methodsavailable for hand and mouse gesture for notebooks, we could use laptop or webcam and hand recognition inourprojectwe couldusethegesturetocontrol themouseandperformbasicoperationssuch as controlling the mouse pointer, selecting and deselecting using the left button and the quick file access function transmission between connected systems using hand movements alone. The system we are implementing is written in

These systems often usemachine learning models to recognize gestures and translate them into mouseactions. They aim to make computing more accessible and hands-free.

However, many still face issues like poor performance in low light and slowresponse time. Despite challenges, gesture-controlled mouse software isbecoming more advanced and user-friendly.

5.PROPOSED SYSTEM

systems connected via a LAN cable. The finished project is "zero cost"hand recognitionsystem thatuses simplealgorithms todo track the hand and hand movements; by assigning an action for each move. But our main focus is primarily on actions such as pointing and clicking, and also defining an action to transfer files between

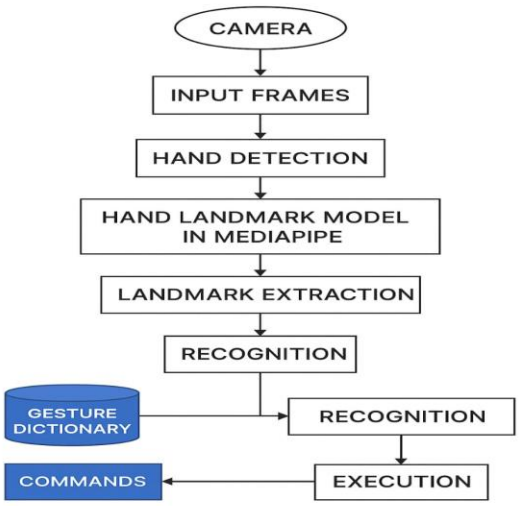
much more responsive python code and is easy to implement because python is a simple

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 3, 2025

language, platform independent, flexibleand portable; thisiswhat was desired when created a program so focused on that the purpose of was to create a virtual mouse and hand recognition system. The system is much more extensible by defining actions for the movement of the needle to perform a specific action. It can be further modified by performing such actions for the whole hand gesture .



6.SYSTEM DESIGN

7.IMPLEMENTATION

The system is implemented in Python 3.8 with the following libraries:

MediaPipe – for real-time hand landmark detection.

OpenCV – for preprocessing and frame visualization.

NumPy – for numerical computation.

PyAutoGUI – for simulating mouse operations.Pynput – for keyboard and multi-selection control.

Key gesture mappings:

Thumb–Index proximity → Left click

Thumb–Middle proximity → Right click

Thumb–Ring proximity → Screenshot

Index–Middle proximity → Select/Deselect

Thumb–Pinky proximity → Multi-selection

8.RESULTS AND DISCUSSION

The system was tested using a standard laptop webcam. Results show:

Accurate performance of cursor navigation, clicking, drag-and-drop, multi-selection, and screenshot capture.

Smooth interaction under proper lighting conditions.

Performance affected mainly by poor lighting or cluttered background.

No external hardware required → making it cost-effective and portable.

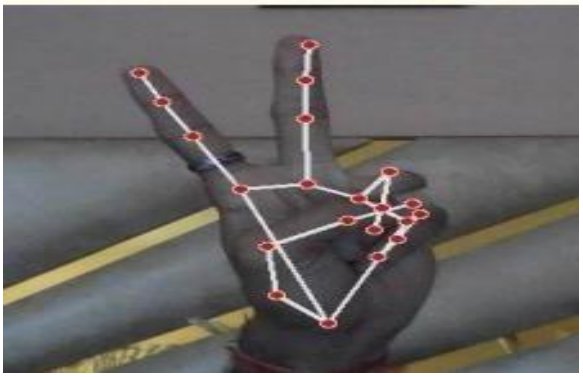
The system demonstrated strong potential for accessibility applications (for users with disabilities), healthcare (contactless device use), and hazardous environments where physical devices may not be safe.

- To perform screenshot

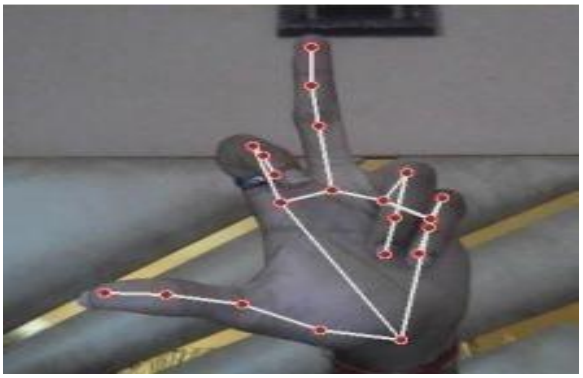
International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

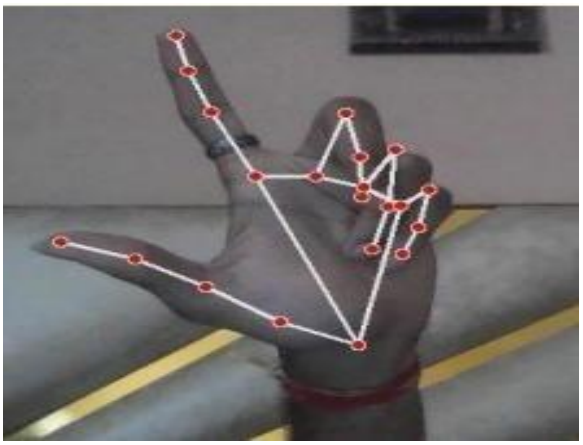
Volume 13, Issue 3, 2025



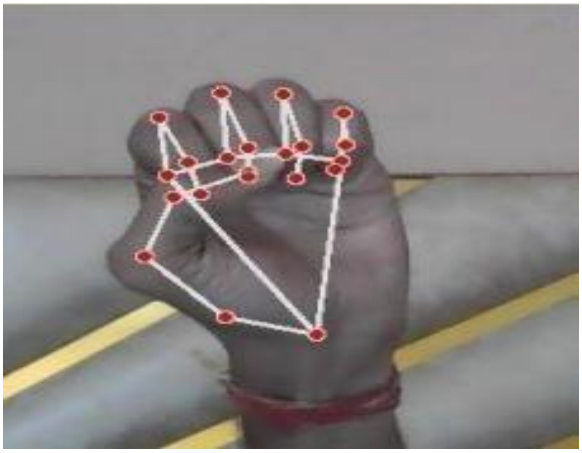
- To Perform Left click



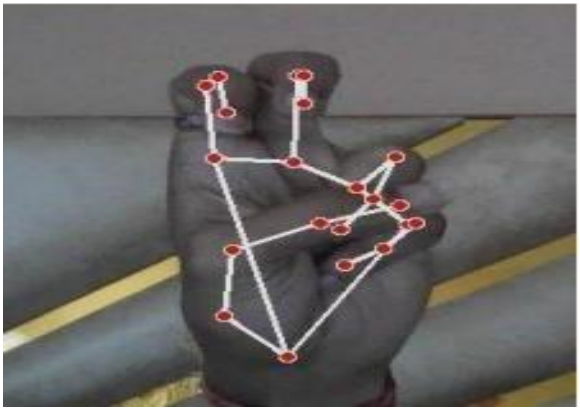
- To Perform Right Button Click operation.



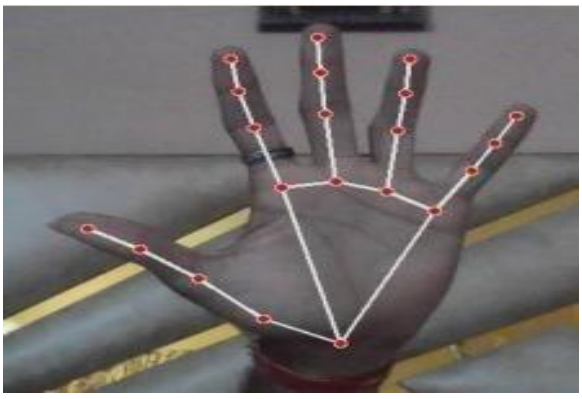
- To perform Drag & Drop Operation



- ToperformMultipleItem Selection.



- ForNoAction/NeutralGesturetobe performedonthe Screen



9.CONCLUSION AND FUTURE WORK

This paper presented a gesture-controlled virtual mouse system using AI-driven hand tracking. By integrating MediaPipe and OpenCV, the system

provided an efficient and cost-effective alternative to physical pointing devices.

Future improvements include:

Incorporating deep learning models for higher accuracy.

Multi-hand gesture recognition.

AR/VR integration.

Mobile device deployment.

This work highlights the potential of gesture-based interfaces for next-generation human–computer interaction.

10.REFERENCES

[1] Sneha U. Dudhane, Monika B. Gandhi, and Ashwini M. Patil, "Cursor Control System Using Hand Gesture Recognition," 2013.

[2] Dung-Hua Liou, Chen-Chiung Hsieh, and David Lee, "A Real-Time Hand Gesture Recognition System Using Motion History Image," 2010.

[3] Quam, "Data Glove-Based Gesture Recognition Model," 1990.

[4] Saurabh Singh, VinayPasi, and PoojaKumari, "Cursor Control using Hand Gestures," IJCA, 2016.

[5] H. McCracken, "Douglas Engelbart: Inventor of the Mouse," Time Magazine, 2013.

[6] T.W. Ng and K.T. Ang, "The Optical Mouse for Harmonic Oscillator," 2005.

[7] D. LeBlanc, H. Hamam, and Y. Bouslimani, "Infrared-based Human–Machine Interaction," Proc. 2nd Int. Conf. ICT, 2006.

These studies indicate the demand for efficient, real-time systems without external markers or costly hardware. The present work addresses these limitations using AI-based vision techniques.