International Journal for Modern Trends in Science and Technology
Volume 12, Issue 04, pages 56-60.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol12issue04.html
DOI: https://doi.org/10.5281/zenodo.19324477

AI Virtual Mouse Using Open CV with Hand Guesters Landmarks

K.V.Supraja, Boganadham Venkata Sai Bhavyasri, Dupati Penchala Divya, Padarthi Annamaiah, Jangam Rohith, Thiruvayipathi Venkata Sai Krishna

Department of CSE-AI, PBR Visvodaya Institute of Technology and Science, Kavali, A.P, India

To Cite this Article
K.V.Supraja, Boganadham Venkata Sai Bhavyasri, Dupati Penchala Divya, Padarthi Annamaiah, Jangam Rohith & Thiruvayipathi Venkata Sai Krishna (2026). AI Virtual Mouse Using Open CV with Hand Guesters Landmarks. International Journal for Modern Trends in Science and Technology, 12(04), 56-60. https://doi.org/10.5281/zenodo.19324477

Article Info
Received: 28 February 2026; Revised: 18 March 2026; Accepted: 22 March 2026.

KEYWORDS

ABSTRACT

Recent advance in artificial intelligence and computer vision have enabled more natural ways of interacting with computers. This paper presents an AI- based Virtual Mouse system that allows users to control cursor operations through hand gestures captured by a webcam. The system processes real-time video input using computer vision techniques and detects hand movements to perform mouse functions such as pointer movement, clicking, dragging, and scrolling without the need for a physical device.

The proposed model is implemented using technologies such as OpenCV for video capture and image processing, NumPy for numerical computations, and PyAutoGUI for executing mouse control actions. The development environment is supported by the Visual Studio platform, which simplifies package management and deployment. By identifying specific hand positions and movements, the system converts gestures into corresponding system commands in real time.

This touch-free approach reduces reliance on hardware, improves hygiene, and provides an affordable solution that requires only a standard camera. The AI Virtual Mouse demonstrates an intuitive, accessible, and efficient alternative to conventional mouse devices, highlighting the growing potential of gesture-based human–computer interaction.

OpenCV (Open Source Computer Vision Library)

OpenCV (Open Source Computer Vision Library) is an open-source library designed for real- time computer vision and image processing applications. In the proposed AI Virtual Mouse system, OpenCV is used to capture live video input from the system's webcam using frame-by-frame processing. Each frame obtained from the camera is preprocessed and forwarded to the

hand detection module for gesture analysis. OpenCV also manages image display, window handling, and real-time rendering, allowing the system to provide immediate visual feedback to the user. Furthermore, OpenCV performs essential preprocessing operations such as

image resizing, color space conversion, and noise reduction, which enhance detection accuracy and system stability

OBJECTIVE

The primary objective of the proposed AI Virtual Mouse system is to develop a touch-free human– computer interaction method that allows users to control mouse functions using hand gestures captured through a webcam. The system aims to replace conventional hardware input devices with an intuitive, vision-based interface that operates in real time.

- To design a contactless mouse control mechanism using computer vision techniques without requiring any physical mouse device
- To detect and track hand movements accurately in real time using advanced hand landmark detection methods.
- To recognize predefined gestures and translate them into mouse operations such as cursor movement, left click, right click, scrolling, dragging, zooming, and screenshot capture.

NEED FOR STUDY

The AI Virtual Mouse allows users to control computer operations using hand gestures captured by a standard webcam. It eliminates the need for physical mouse devices, reducing hardware dependency and maintenance costs.

This touch-free approach also improves accessibility and individuals with mobility

- Requires physical contact with hardware devices
- Limited accessibility for users with disabilities
- Prone to mechanical wear and malfunction
- Inconvenient in hygienic or touch- restricted environments.

With rapid advancements in Artificial Intelligence, Computer Vision, and hand-tracking technologies, gesture-based systems can interpret human movements in real time using a standard webcam. Techniques such as hand landmark detection and image processing enable accurate recognition of finger positions and gestures, allowing computers to be controlled without traditional input Enables touch- free computer interaction

- Supports users with limited mobility
- Reduces dependence on external hardware
- Improves hygiene and safety in shared environments.

EXISTING SYSTEM

The existing system for computer interaction mainly depends on physical input devices such as the mouse and keyboard. These devices allow users to control the cursor, perform clicks, scroll, and execute commands through direct hand contact. Although modern mice have evolved into optical and wireless versions with improved accuracy and convenience, they still require a physical surface and manual operation to function effectively.

However, traditional input devices have several limitations. They are prone to mechanical wear, connectivity problems, and hardware failures over time. In addition, they may not be suitable in environments where space is limited or where touch- free interaction is required for hygiene and safety. Conventional systems also pose accessibility challenges for individuals with physical disabilities, highlighting the need for alternative methods of human–computer interaction.

Disadvantages

- Sensitive to lighting and background conditions.
- Limited gesture recognition capability
- Requires manual calibration.
- Moderate accuracy.
- Requires continuous physical contact with input Device
- Slow response time (high latency.
- Performance Varies for different users.
- Not robust for real-time complex environments
- Restricted portability due to dependence on hardware.

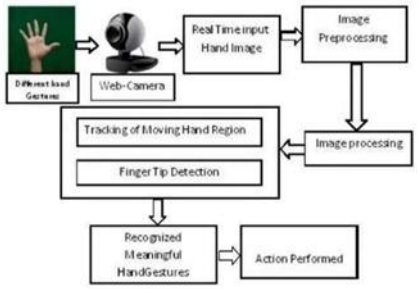
PROPOSED SYSTEM

The proposed system is an AI-based Virtual Mouse that allows users to control the computer using hand gestures, eliminating the need for physical mouse devices. It uses a webcam to capture real- time video and Media Pipe to detect and track hand landmarks like fingertips and joints. The system recognizes gestures such as cursor movement, left/right click, scrolling, and optional system controls like volume or brightness adjustment. Detected gestures are mapped to computer

actions using PyAutoGUI, providing smooth, accurate, and touch-free interaction. The system also provides visual feedback to the user for gesture confirmation, ensuring intuitive and reliable control. Compared to conventional systems, this approach offers a natural, hygienic, and innovative way.

SYSTEM ARCHITECTURE

SYSTEM ARCHITECTURE:



MODULES

- Input Capture Module
- Hand Detection Module
- Gesture Recognition Module
- Cursor Control Module
- Mouse Action Module
- System Control Module
- Output Module

A. Input Capture Module

This module captures real-time video from the webcam using OpenCV. It continuously provides frames for processing and ensures smooth and accurate gesture detection. The quality of capture directly affects the system's performance

B. Hand Detection Module

Detects and tracks hand landmarks like fingertips, joints, and palm using Media Pipe. It filters out background noise and isolates the hand precisely. Coordinates are sent to the gesture recognition module for further processing.

C. Gesture Recognition Module

Interprets hand positions as gestures such as cursor movement, left/right click, and scrolling. Uses the landmark coordinates to identify gestures accurately in real time. Recognize guestures are sent to control modules for action

D. Cursor Control Module

Maps hand movements to screen coordinates for smooth and responsive cursor control. Applies smoothing techniques to reduce jitter. Provides a natural and intuitive pointer movement for the user.

E. Mouse Action Module

Executes mouse operations like click, double-click, scroll, and drag-and-drop using PyAutoGUI. Connects the recognized gestures to actual system actions. Ensures reliable and seamless control of the computer.

F. System Control Module

Allows gesture-based control of system functions like volume, brightness, window management, and screen lock. Extends the AI Virtual Mouse functionality beyond basic mouse operations. Enhances overall usability and convenience.

G. Output Model

Executes all recognized gestures and cursor actions on the system in real time. Ensures smooth touch- free control and completes the human- computer interaction cycle. Makes the system practical and user-friendly.

H. Advantages

- **Touch-Free Operation** – Users can control the computer without physically touching a mouse, reducing contact and improving hygiene.
- **Accessibility** – Useful for people with physical disabilities or limitations in using a traditional mouse.
- **Multi-Functional** – Supports cursor control, clicks, scrolling, and optional system controls like volume and brightness adjustment.
- **Portable and Cost-Effective** – Works with a standard webcam, eliminating the need for additional hardware.
- **Real-Time Response** – Provides fast and accurate gesture recognition for smooth operation.
- **Enhanced User Experience** – Visual feedback allows users to see detected gestures, ensuring correct actions every time.
- **Innovative Technology** – Integrates AI, computer vision, and gesture recognition for modern

human-computer interaction.

I. Hardware Requirements

- Processor : Intel i5
- RAM : Minimum 8GB
- GPU : NVIDIA GTX 1080 or higher (for
- deep learning models)
- Storage : SSD with at least 512 GB
- Display : High-resolution monitor (for image visualization)

J. Software Requirements:

- Operating System : Windows 10/11,
- Ubuntu (Linux), or macOS
- Programming Language : Python
- Libraries & Frame works : TensorFlow
- / PyTorch, OpenCV, NumPy, Pandas,
- Matplotlib
- Technologies : Computer Vision, Media
- pipe,Open CV\
- Environment : VS Code

TECHNIQUES USED IN THE PROJECT

Computer Vision (OpenCV) – Captures and processes real-time video frames from the webcam for hand tracking.
Hand Landmark Detection (Media Pipe) – Detects and tracks hand landmarks such as fingertips, joints, and palm for gesture recognition.
Gesture Recognition – Interprets the hand landmarks to identify gestures like cursor movement, left/right click, scrolling, and system controls.
Mouse Control (PyAutoGUI) – Maps recognized gestures to computer actions such as clicks, scrolling, dragging, and system commands.
Coordinate Mapping & Smoothing – Converts hand movement into screen coordinates and applies smoothing algorithms to reduce cursor jitter.

CONCLUSION

The AI Virtual Mouse using OpenCV and hand gestures presents an innovative approach to human– computer interaction. By replacing a traditional mouse with simple hand movements captured through a webcam, the system creates a more natural and engaging user experience. It demonstrates how computer vision and Artificial Intelligence can work together to understand human gestures and translate them into meaningful digital actions. This project also proves that advanced technology does not always require complex hardware. With the help of OpenCV and gesture recognition techniques, the system can perform cursor movement, clicking, and scrolling efficiently. It highlights the practical implementation of AI concepts and shows how they can be applied to solve real-world problems in a simple and effective manner.

.

FUTURE ENHANCEMENT

In the future, the AI Virtual Mouse system can be improved by supporting gestures from both hands, allowing more complex and versatile controls. Users could be given the option to create custom gestures for personalized actions. Integration with voice commands would make the system more interactive and efficient. Additionally, incorporating 3D gesture detection using depth sensors could improve accuracy and responsiveness. The system can also be adapted for specific applications like gaming, presentations, or media control, and expanded to work on mobile and AR/VR devices. Further improvements in gesture recognition under different lighting and background conditions will make the system more reliable and user-friendly.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] OpenCV, "Open Source Computer Vision Library," OpenCV Foundation, 2021. Available: https://opencv.org

[2] Google MediaPipe, "MediaPipe Framework: A Cross-Platform Framework for Building Perception Pipelines," Google Research, 2021. Available: https://mediapipe.dev

[3] IEEE, "Hand Gesture Recognition using Deep Learning Techniques," IEEE Access Journal, Vol. 9, 2021, pp. 1–10.

[4] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 37, No. 3, 2021, pp. 311–324.

[5] Springer, "Advances in Computer Vision and Pattern Recognition," Springer Publications, 2022. Available: https://link.springer.com

[6] Elsevier, "Artificial Intelligence and Computer Vision Applications," ScienceDirect Journals, 2022. Available: https://www.sciencedirect.com

[7] ACM, "Hand Gesture Recognition for Human– Computer Interaction," ACM Digital Library, 2023. Available: https://dl.acm.org

[8] R. S. Kumar, P. Sharma, “Real-Time Virtual Mouse System using Hand Gesture Recognition,” International Journal of Computer Applications, Vol. 184, No. 12, 2022, pp. 15–20.

[9] A. Verma, S. Gupta, “AI-Based Virtual Mouse using Computer Vision,” International Journal of Engineering and Advanced Technology (IJEAT), Vol. 11, Issue 3, 2023, pp. 55–60.

[10] K. Patel, R. Shah, “Touchless Human–Computer Interaction using Hand Gesture Recognition,” International Journal of Computer Science and Information Security (IJCSIS), Vol. 21, No. 2, 2023, pp. 45–50.