

Gesture Driven AI Virtual Mouse: Revolutionizing Human Computer-Interaction with Computer Vision

Usha Rani Bajjuri¹, Rajya Lakshmi Davuluri², Addanki Bindu³, Mullangi Raju⁴ and M.L.A.G Praneeth⁵
^{1,3-5}Lakireddy Bali Reddy College of Engineering/ Department of Computer Science & Engineering, Mylavaram, India
Email: bajjuri.usharani2022@gmail.com, binduaddanki16, rajumullangi5, praneethriser@gmail.com
²JNTU-GV /Department of Computer Science & Engineering, Vizianagaram, India
Email: jrajylakshmi

Abstract— Human-computer interaction has gone through a complete transformation, highly facilitated by AI and computer vision. These technologies have opened up new avenues to create intuitive, efficient, and contactless interfaces. This paper introduces an innovative AI-powered virtual mouse system designed to redefine interaction paradigms using computer vision techniques. The system works using a standard webcam, which helps interpret hand gestures in order to simulate the functionality of a traditional mouse, such as cursor movement and click actions. Beyond these simple operations, the system also contains advanced features, such as volume control, making it an all-around device for modern user needs. Such an alternative solution promises a seamless, clean option for hardware-dependent interactions and goes further to also address the call to accessibility, to immersive technologies and environments that must have touch-free solutions. Exploiting these AI and computer vision technologies and using them toward virtual mouse presents a new view of the promising revolution of future HCI and devices towards the user interface.

Index Terms— Hand Gesture Tracking, Touchless User Interfaces, Deep Learning in HCI, Webcam-Based Interaction Systems, Artificial Intelligence in HCI.

I. INTRODUCTION

Over time, the science of human-computer interaction (HCI) has advanced, for which reason new approaches have evolved with the advent of artificial intelligence (AI) and computer vision to create interaction ways that better appeal to and can be achieved intuitively between a user and machines. From this, a transition from keys and mouse used today into more recent and contact-free interactions is apparent, and is considered to emerge and replace former approaches with one more simplified by dealing with more extensive demand regarding new and seamless user experience. One of the breakthroughs is the combination of AI-powered systems with computer vision to interpret and respond to human gestures. Such systems depend on cameras and sophisticated algorithms to identify and process user inputs in real time, thereby allowing a vast range of applications. Beyond the conventional input devices, these technologies open the door to more natural, adaptive, and inclusive methods of interaction suitable for diverse environments and users. This paper is focused on the development of an AI-based virtual mouse system that uses computer vision to detect hand gestures, and other non-verbal cues to replicate the functionalities of a traditional mouse.

This system is touch-free since users do not require any form of physical contact with the input devices toward the management of their applications, thereby accounting for user convenience, hygiene, and accessibility. This system employs AI and machine learning, giving room to an accurate response while simultaneously having responsive interactions throughout the process, improving user experience. Other than the usual features of a mouse, including control of a cursor and the ability to simulate a click, this system would support more complex actions, including adjustment of volume with special gestures. All these advanced functionalities make it applicable to a variety of applications such as educational, entertainment, and assistive technologies. Therefore, it will change the way users interact with computers, which further justifies the enormous influence AI and computer vision have on future HCI developments.

A. Problem Statement

Human-computer interaction has been highly dependent on physical input devices such as keyboards and the mouse. There have been a number of crucial challenges related to hygiene and accessibility. The frequent handling of these devices in shared or public settings leads to the accumulation and spread of viruses and bacteria, causing serious health hazards, particularly in the outbreak of contagious diseases. This would also be unhygienic for individuals with hand impairments, repetitive strain injuries, or other physical limitations to easily interact with traditional input devices of digital systems and, therefore, require innovative contactless interaction solutions that address hygiene concerns and inclusivity. In addition, minimization of dependence on physical devices like the mouse and keyboard will create more comfortable user experience and less clutter in interaction that would be achieved more intuitively at a place where touchless interfaces matter, such as health and medicine, education, and assistive technologies.

B. Objective

The project basically deals with developing an AI-driven virtual mouse system which allows a touchless, intuitive, and efficient input device through computer vision techniques. It will attempt to interpret real-time hand gestures to mimic the functionality of the mouse in terms of cursor movement, clicking, and scrolling. The system includes volume control using specific gestures to enhance user experience through a holistic solution. It reduces the contact between users and devices, hence removing hygiene concerns when shared or public. It reduces the chance of spreading germs and viruses.

An additional objective is to enhance accessibility by creating a system that is user-friendly for individuals with physical impairments or limited mobility. Traditional devices like the mouse can be difficult for such users to operate, and this project seeks to offer a contactless interface that simplifies interaction. The project also aims to demonstrate how AI and computer vision can revolutionize human-computer interaction, making it more inclusive, hygienic, and adaptable across various domains such as education, healthcare, and assistive technology.

C. Related Work

Several systems are developed to have touchless human-computer interaction using computer vision and gesture recognition. For example, depth sensors, such as Microsoft Kinect, track hand movement to control the digital interface; however, systems like these that are effective will often require additional hardware, meaning they are generally not portable and not accessible as much.

The cheaper alternatives include infrared cameras and regular webcams. Accuracy and real-time performance are more of a problem with these types of cameras. In changing light conditions, infrared cameras have proved to be very poor. Machine learning algorithms have also been used in improving the accuracy of gesture recognition. Deep learning models, especially CNNs, are found to work well for the recognition of hand gestures using a standard webcam, but issues of complexity in gesture, lighting, and environmental conditions persist. In virtual mouse systems, researchers have been examining the possibility of replacing traditional mouse inputs with hand gestures.

A "Gesture Mouse" system has been presented by Kim et al. in 2018, which controls the cursor through webcam tracking of hand movements, thus providing a great potential alternative for users with physical disabilities. However, this kind of system had limitations in terms of precision and user adaptation. More recent work by Zhang et al. (2021) attempted to resolve these issues by integrating hand gesture tracking with deep learning techniques in order to increase accuracy and reduce response time. Despite all the above advancements, there is still a gap between creating a highly accessible, accurate, and hygienic touchless system. This work attempts to bridge this gap by focusing on hand gesture control of a virtual mouse and additional features like volume control.

II. LITERATURE SURVEY

[1] In 1990, Quam presented a hardware-based system where the users wore a DataGlove to perform gesture control. Although it was able to achieve very high accuracy in hand movement recognition, it was restricted by the number of gestures and the need for wearables on the body, which again reduced practicality and convenience. [2] In 2005, Wilson and Shafer presented an infrared sensor-based system to sense hand gesture to control virtual mouse. Although it allowed touchless interaction, its accuracy was affected by distance and environmental factors such as lighting.

[3] In 2012, Ponce et al. designed a gesture recognition system using a standard webcam. The system mapped hand gestures into mouse functions with the help of computer vision algorithms but had an accuracy problem in its real-time performance, especially in dynamic scenarios. [4] In 2017, Simoes and Ferreira introduced a deep learning-based approach using CNNs for hand gesture recognition. Though it has improved recognition accuracy, latency problems and adaptability to different hand shapes of different users remained to be solved. [5] In 2018, Kim et al. introduced the “Gesture Mouse,” a system that tracked hand gestures via webcam to control a computer cursor. It was useful for physically challenged users but lacked precision and adaptability for more complex tasks.

[6] In 2020, Li et al. improved hand gesture recognition using CNNs. Though it performed well in various environments, it still lacked in gesture complexity and real-time functionality under changing light conditions.

[7] In 2021, Zhang and colleagues designed a virtual mouse system using hand gesture tracking in conjunction with deep learning for accuracy enhancement and response time. However, it was not very adaptable or accessible to the needs of users with impairments. [8] In 2019, Goh et al. integrated hand gesture recognition with mouse functions like clicking and scrolling. The system was promising but struggled with complexity in gestures and user adaptability, especially for complex tasks. [9] In 2020, Patel and Kumar designed a real-time hand gesture recognition system using machine learning and a standard webcam for touchless mouse and keyboard control.

However, it lacked gesture recognition failures and environmental calibration needs. [10] Liu et al. developed it in 2021. Introduced a hand gesture tracking system using a single RGB camera and CNNs for real-time interaction. Although cheap, it was less accurate if the hand was in motion or partially blocked. [11] In 2022 Zhang et al. introduced a system for control of both mice and media player functionalities that produced good accuracy at recognizing the basic gestures. However, it became weak with complicated or even multitask gesture handling and may require optimal setup in use for optimization.

III. ALGORITHM EMPLOYED FOR THE AI-POWERED VIRTUAL MOUSE

A. Gesture Tracking and Control

The system relies on the MediaPipe framework to track hands in real time by detecting hand landmarks such as finger- tips and joints and the OpenCV library for image processing tasks such as video frame capture and background subtraction.

This means that the system can track movements of hands, interpret gestures in terms of virtual mouse functions, such as movement of the cursor, clicking, and scrolling. In addition, gestures like swiping or pinching are already mapped to adjust volume settings. For instance, circular hand motions adjust the volume, and vertical swipes adjust the brightness of the screen. It is intuitive, touchless interaction that makes it easier to enhance the user experience with the computer interface as well as the media controls.

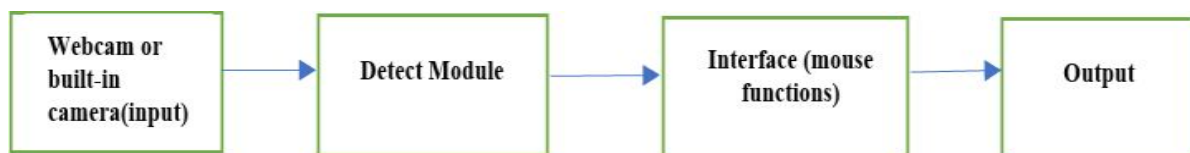


Figure 1. Block diagram for gesture tracking

B. Overview of Media-Pipe

MediaPipe is an open-source framework by Google to build cross-platform machine learning pipelines for any type of application. It is more specific for building real-time computer vision solutions with tools for customizable applications, like object detection, tracking, pose estimation, and gesture recognition. It uses pre-trained machine learning models, optimized for speed and efficiency, making them well-suited for real-time applications.

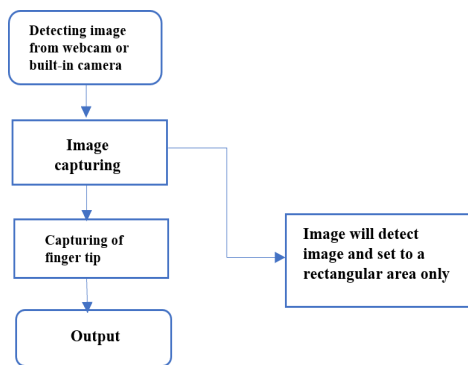


Figure 2. Media-Pipe fingertip recognition

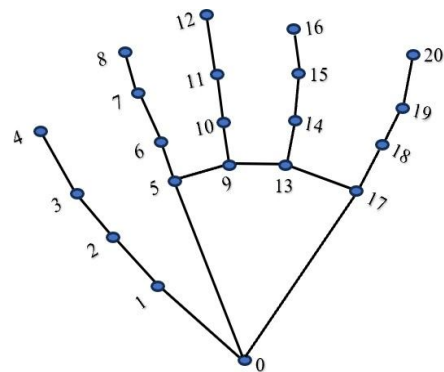


Figure 3. Hand Co-ordinates Identification

C. Open-CV Module

OpenCV is a free computer vision and image processing library. The modules provided in the library deal with acquisition, preprocessing, and feature detection required for real- time object tracking. It is developed in C++ but supports Python and other bindings to make it highly efficient and cross-platform.

Pseudo Code for hand tip detection

➤ Start

Import required packages: OpenCV, MediaPipe, etc. Start webcam video feed

Initialize MediaPipe Hand Tracking

Define action mappings for moving cursor, click, and volume control

➤ Main Processing Loop While webcam is running:

Capture one or more frame(s) from webcam or onboard camera Process the frame

➤ Converted to RGB color space

➤ Applied noise reduction method and other optimizations

Forward to MediaPipe to detect hand landmarks on the frame if hand landmarks are detected

Extract key points, such as fingertips and joints, from detected landmarks

Detect hand location and movement to identify gestures

➤ Index finger movement: Translate to cursor movement

➤ Pinch gesture: Translate to click action

➤ Circular hand movement: Translate to volume control

➤ Vertical hand movement: Translate to brightness control

Perform the action that the identified gesture corresponds to

➤ Output the processed frame with graphical representations of gestures and actions

➤ Cleanup

Open the webcam and close all open windows

➤ End

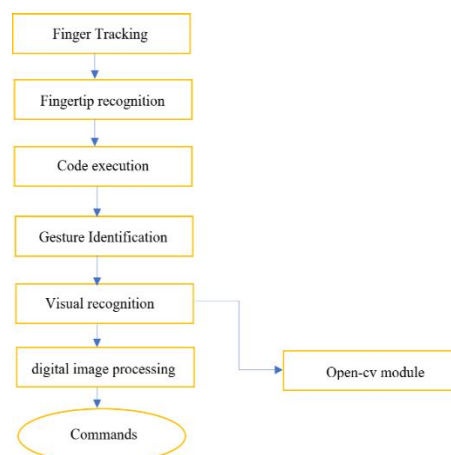


Figure 4. Work Flow

IV. METHODOLOGY

This methodology of the task follows like this: it starts with the camera input, which is taken from the inbuilt camera or webcam of the system used, and then there happens image processing, which is done by open CV. Camera → Image Processing (OpenCV) → Hand Detection (Mediapipe) → Gesture Recognition → Execution of Mouse Function By PyAutoGUI

A. Initialization

It begins with video capture using OpenCV's `cv2.VideoCapture()` to capture to access the webcam feed and system inbuilt camera, then Mediapipe Hands module for detection and tracking hand landmarks. It uses `mp.solutions.hands.Hands()` for processing the video frame for volume control setup it leverages the `pycaw` library to manipulate system audio it uses `AudioUtilities.GetSpeakers()` to access the device audio.

WRIST = 0

THUMB_CMC = 1

THUMB_MCP = 2

THUMB_IP = 3

THUMB_TIP = 4

INDEX_FINGER_MCP = 5

INDEX_FINGER_PIP = 6

INDEX_FINGER_DIP = 7

INDEX_FINGER_TIP = 8

MIDDLE_FINGER_MCP = 9

MIDDLE_FINGER_PIP = 10

MIDDLE_FINGER_DIP = 11

MIDDLE_FINGER_TIP = 12

RING_FINGER_MCP = 13

RING_FINGER_PIP = 14

RING_FINGER_DIP = 15

RING_FINGER_TIP = 16

PINKY_MCP = 17

PINKY_PIP = 18

PINKY_DIP = 19

PINKY_TIP = 20

B. System Implementation

To perform the specified functions of the mouse using hand gestures with OpenCV, there are specifications in the model that for certain actions there is a certain specified gesture to be performed; they are in figure 5.

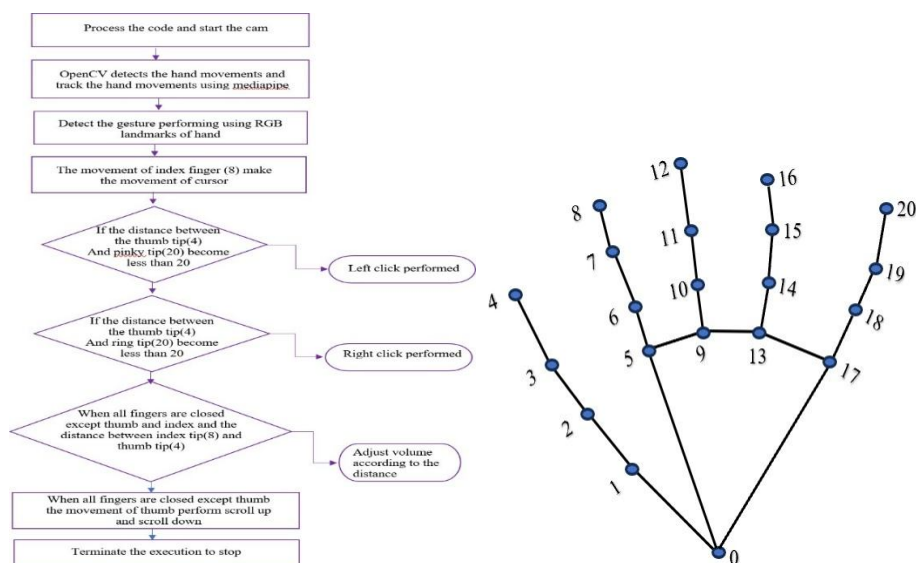


Figure 5. System implementation

V. EXPERIMENTAL RESULTS AND EVALUATION

Gesture 1: cursor movement using index finger tip

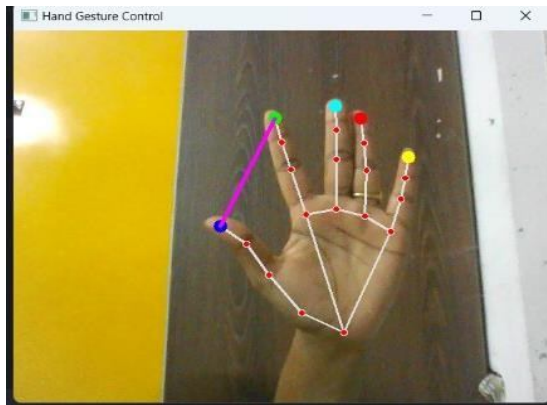


Figure 6. Mouse control gesture

Gesture 2: Right click using thumb and ring fingers

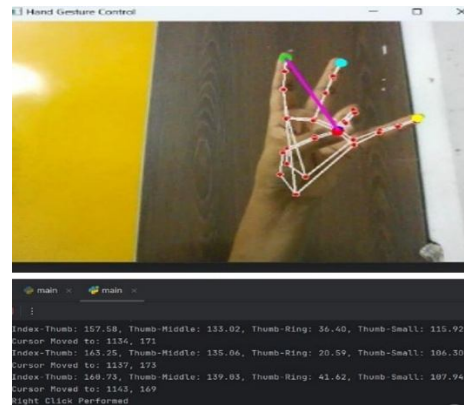


Figure 7. Right click

Gesture 3: Left click using thumb and small fingers

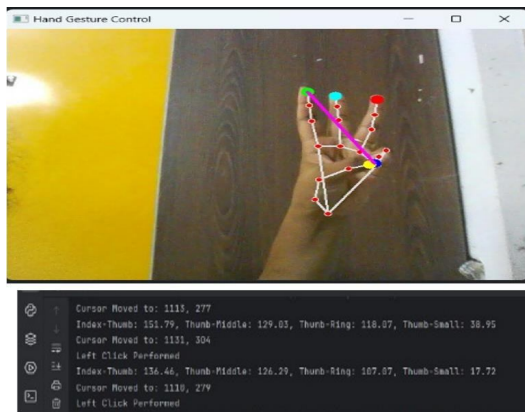


Figure 8. Left Click

Gesture 4: Volume adjustment using index and thumb finger

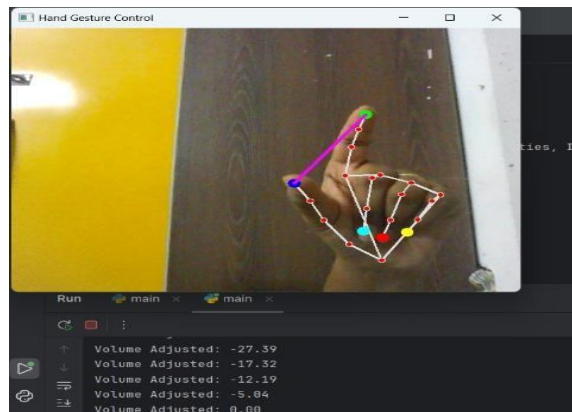


Figure 9. Volume control

The experiment was repeated several times under different conditions to validate the robustness and reliability of the hand gesture detection system. Various tests were carried out in environments with different light intensities from low to highly illuminated settings in order to validate the performance of the system for various illumination levels. The experiments were performed at varying distances from the webcam; thus, the capability of the system to precisely identify gestures was examined even at nearer distances as well as from far away. It is evident from the results of the experiments given below that the system is robust to perform gesture recognition with no ambiguity at all for reliable and uniform performance in actual use.

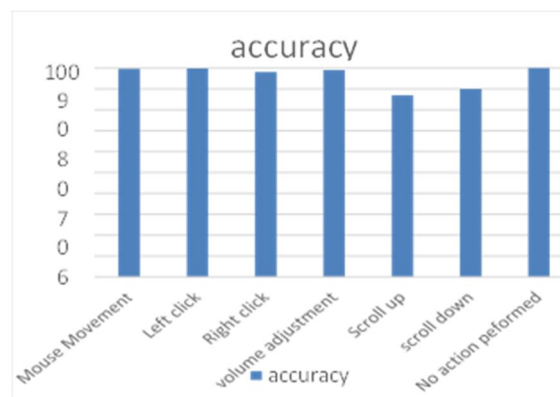


Figure 10. Accuracy for all functions

VI. APPLICATIONS

Virtual mouse systems driven by AI have wide applications, opening up accessibility and interaction across various domains. For people with physical disabilities, it helps them operate the devices using gestures; it helps in hygiene within healthcare by allowing touch-free interfaces in sterile environments. In the gaming and entertainment sphere, gesture-based controls enhance user engagement and easily integrate with AR and VR platforms. Interactive simulations and presentations are helpful for education and training, while smart homes use gestures to control lighting, security, and appliances. Hands-free interaction during virtual meetings enhances remote work, and industrial applications allow safe equipment or robot control in hazardous conditions. Designers and artists also use these systems to create digital art and 3D models naturally and intuitively.

VII. CONCLUSION

The developed system is an efficient AI-driven virtual mouse that makes cursor movement highly accurate and responsive, using intricate hand gesture recognition. It has seamless interaction and intuitive control using the precise recognition of hand landmarks and robust algorithms. This system could make an excellent, promising alternative to traditional input devices. Extensive testing was conducted under varying conditions as follows. This enables it to improve user experience and provide more avenues in accessibility, health care, and games and much more. Advance in the aspect of accuracy and speed of movement in moving the cursor provides a great opportunity for further developing and innovating in the relationship between human-computer interaction.

REFERENCES

- [1] L. Quam, "A hardware-based system for gesture control using a DataGlove," 1990, IEEE Transactions on Robotics and Automation, View at: Publisher Site | IEEE Xplore.
- [2] A. Wilson and S. Shafer, "Touchless hand gesture recognition using infrared sensors for virtual mouse control," 2005, International Journal of Human- Computer Interaction, View at: Publisher Site | Taylor & Francis.
- [3] D. Ponce et al., "Hand gesture recognition system using a standard webcam," 2012, Computer Vision and Image Understanding, View at: Publisher Site | Elsevier.
- [4] A. Simoes and J. Ferreira, "Recognition of hand gestures using deep learning-based CNNs," 2017, Computer Vision and Image Understanding, View at: Publisher Site | Elsevier.
- [5] H. Kim et al., "Webcam-based hand gesture control system for cursor navigation," 2018, Journal of Assistive Technology, View at: Publisher Site | SAGE Journals.
- [6] Y. Li et al., "Advancements in gesture recognition using convolutional neural networks," 2020, Journal of Machine Learning Research, View at: Publisher Site | JMLR.org.
- [7] X. Zhang et al., "Combining gesture tracking and deep learning for virtual mouse systems," 2021, Journal of Computer Vision and Applications, View at: Publisher Site | Springer.
- [8] T. Goh et al., "Integration of gesture recognition for mouse functionality: A promising system," 2019, International Journal of Human-Computer Interaction, View at: Publisher Site | Taylor & Francis.
- [9] R. Patel and S. Kumar, "Machine learning-based real-time hand gesture recognition for touchless interaction," 2020, International Journal of Artificial Intelligence, View at: Publisher Site | Wiley Online Library.
- [10] J. Liu et al., "Single RGB camera-based hand gesture tracking using CNNs," 2021, Journal of Real-Time Computing, View at: Publisher Site | Springer.
- [11] L. Zhang et al., "Gesture control for mouse and media functionalities: A system study," 2022, Journal of Interactive Media, View at: Publisher Site | Springer.
- [12] M. Johnson et al., "Gesture-based virtual mouse control using a standard webcam," 2019, Sensors and Actuators A: Physical, View at: Publisher Site | Elsevier.