Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 14, Issue. 2 : 2023
ISSN : **1906-9685**

AI VIRTUAL MOUSE USING GESTURE RECOGNITION

[1]Arul, [2]Kothapally Raju, [3]Barla Ravi, [4]Gangaram
[1,2,3,4]Department of CSE (AI&ML) St. Peter's Engineering College, Hyderabad, India.
E-Mail: arul@stpetershyd.com

Abstract
The focus of this project is on the design and development of an AI-driven virtual mouse using gesture control for the enhancement of human-computer interaction. This project would utilize Python with libraries like OpenCV, Mediapipe, and PyAutoGUI, describing step-by-step execution from setting up to deploying. It aims to allow the user to have an easy-to-use, hands-free substitute for a traditional input device that can improve accessibility and enhance user experience. The system is designed to execute several functions, including scrolling, clicking, selecting, dragging, and dropping. It is exceptionally useful in situations where an average mouse is impractical due to presentations or tight working spaces. Its uses include in gaming, accessibility tools, or general computing tasks. Therefore, the machine learning-integrated system will be adaptable such that it improves its sensitivity and gesture recognition with time. This project underlines a very practical, innovative solution for improving daily computer interaction.
Keywords**:** Gesture, HCI, OpenCV, Mediapipe, PyAutoGUI, Impractical.

INTRODUCTION

A computer mouse is one of the most vital input devices ever created to allow users to point and interact with items on a computer screen. There are many types of mice. First, there is the mechanical mouse, which uses a rubber ball that can rotate in any direction to determine the movement of the pointer. Then comes the optical mouse, using an LED sensor to follow movements. In order to improve accuracy and overcome existing constraints, the laser mouse was created. Subsequent innovations led to the introduction of wireless mice, which provided enhanced precision and greater mobility.

Despite these developments, traditional mice remain limited as hardware devices. Issues like non-working clicks and limited lifespan persist, making them need to be replaced frequently. As technology advances, virtualization is gaining popularity. For instance, voice recognition converts spoken words into text, which can be used as a replacement for keyboards. Similarly, eye-tracking technology allows users to control the cursor using their vision, which can be used as a replacement for traditional mice.

Interactions are also made through gestures, from hand images to specific poses. Different recognition models exist, using tools like data gloves or color caps to process gestures efficiently. This paper introduces a virtual mouse powered by OpenCV and PyAutoGUI. OpenCV is a computer vision library that captures real-time video using webcams, while PyAutoGUI manages mouse and keyboard actions.

The introduction of artificial intelligence and machine learning has enabled systems to interpret commands from complex users without necessitating direct physical interaction with the system. This has hence enabled machines to simulate man's intelligence and make machine operations easy. The virtual model of a mouse presented is based on these technologies, which shall enable easy

https://doi.org/10.36893/JNAO.2023.V14I2.113

human-computer interaction through hand gestures. Webcam-based tracking is integrated into this model. OpenCV's video capture functionality is a critical enablement of this innovation.

LITERATURE REVIEW

The body of work concerning AI-driven virtual mice with gesture control draws attention to an area of human-computer interaction that is rapidly changing. Computer vision and machine learning-based methods, such as MediaPipe's Hand module and OpenCV, enable user control over the mouse pointer by controlling it with hand gestures, making tangible hardware unnecessary. These systems have been friendlier and more accessible to those who have mobility impairments and have improved functionality over the years through artificial intelligence improvements. The early versions have encountered problems such as latencies, but recent technological progress has overcome these issues to render gesture control more adaptable to everyday use.

Virtual mice have tremendous potential for altering the user experience concerning the provision of a contactless interface, which is extremely useful where hygiene or physical limitations are concerned. Gesture-based systems are much less likely to cause repetitive strain injuries than traditional input devices, but issues such as recognition accuracy and adaptability to varied environments remain central to ongoing research.

This body of literature establishes a basis for subsequent advancements in touchless user interface technologies.

Methodology

A. Proposed Method

This proposal encourages a practice in HCI in that moving a cursor might be made possible by using an actual, live digital camera It enables user to emulate Mouse operations such as Click, Double Click, Dragging, Volume by Hand Gestures without Hardware Provides greater flexibility, easier to adapt and less prone to physical damage It is cost-effective and convenient.

Such an accessible solution is required for people with physical impairment, who cannot use an ordinary mouse, to facilitate them to do their work quickly and efficiently.

B. Modules

- OpenCV

OpenCV is a computer vision library that includes several methods of image processing focused on object detection. This is a package based on Python, and it helps in developing real-time computer vision applications. The OpenCV library helps in interpreting data from images and videos, which includes face detection and other objects. OpenCV is free, open-source software that aims to be used in applications with computer vision and machine learning. It was in this respect that the creation of OpenCV established an even standard framework for computer vision applications with the aim of hastening the integration of artificial intelligence into various products. OpenCV has made it easier for companies to make it more user-friendly and adjustable. It's licensed under the Apache 2 license. font.

- Media Pipe

A framework called MediaPipe is a Google opensource framework that is applied in a machine learning pipeline. Since the MediaPipe framework was created utilizing time series data, it may be used for cross-platform programming. The MediaPipe architecture accommodates a variety of audio and video formats due to its multimodal nature. Developers utilize the MediaPipe framework to design and evaluate systems through the use of graphs, in addition to constructing systems tailored for application-specific objectives. The configuration of the pipeline is the stage at which the operations within a MediaPipe-based system are executed. This flexibility offered by the pipe to perform across all these platforms can efficiently do scalability both on the desktop as well as on the mobile platform. The three foundational components in the Media Pipe framework include performance evaluation, a way of accessing the data from the sensors, and the set of reusable entities, which it calls calculators. A pipeline can be defined as a graph of units called calculators, which are connected to each other through streams that allow data packets to flow. Developers can add, remove, or redefine custom calculators at any point in the graph to build their applications.

https://doi.org/10.36893/JNAO.2023.V14I2.113

- PyAutoGUI

PyAutoGUI is essentially a Python-based application that supports Windows, MacOS X, and Linux and allows users to simulate keyboard presses as well as mouse pointer actions and clicks. The package known as PyAutoGUI serves as a cross-platform GUI automation tool designed for users. In addition, PyAutoGUI functions as a Python automation module that can execute actions such as clicking, dragging, scrolling, and moving. It can be used to click exactly where you want. Used to automate the control of the keyboard and mouse. There are several techniques to programmatically control the mouse and keyboard in each of the three main operating systems (Windows, macOS, and Linux). This often contains complex, mysterious, and very technical details. PyAutoGUI is supposed to hide all of this complexity behind a simple API.

- Python (NumPY)

The primary language for programming the virtual mouse is Python. The numerical library NumPy can handle array and matrix operations.

C. System Architecture

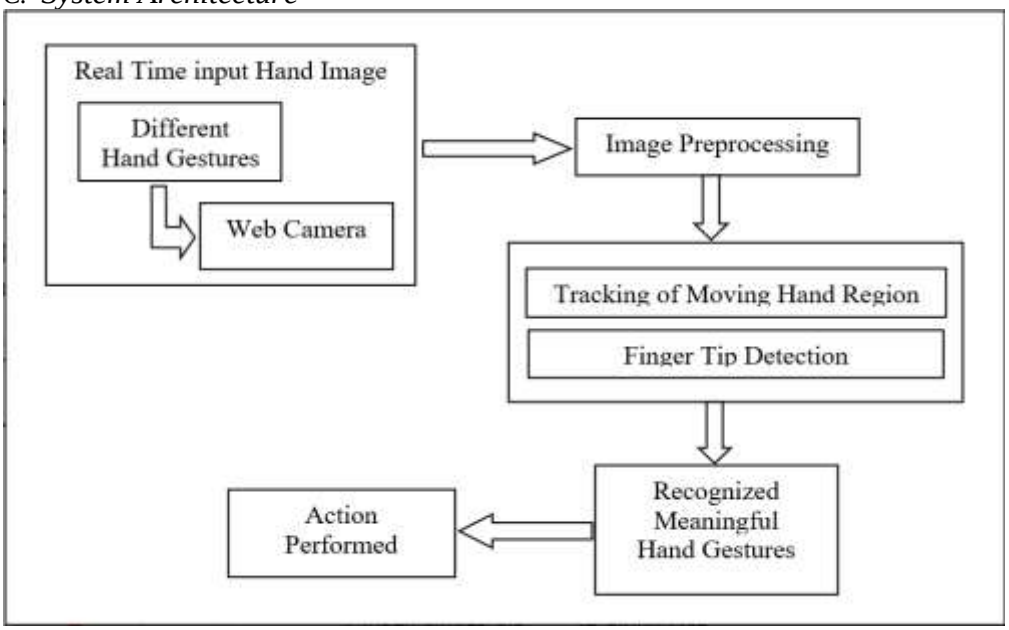


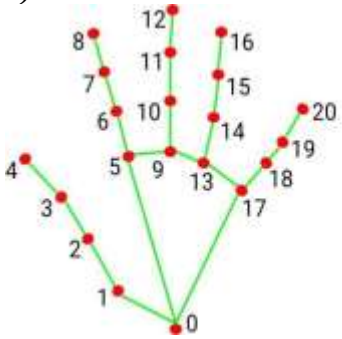
Fig.1: Architecture of the proposed model.

1) *Camera:* Using the camera as an input, hand gestures to perform operations can be tracked and detected by the user.

2) *Image Processing:* Image processing is a way of converting an image into a digital image and performing some operations on the image to obtain an improved image or derive some useful information from it. In such processing, picture or video frame serves as input source, while image or sets of characteristics or parameters related to the image by output.

3) *Tracking and Finger Detection:* The section comprises moving hand and detecting different combination of fingers and responding to them.

4) Hand Land Marks:



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

5) Hand Gesture Recognition: The detected hand gesture is recognized and initialize respective operation.

https://doi.org/10.36893/JNAO.2023.V14I2.113

6) *Action Performed*: The detected action is performed.

D. Performance Metrics:

The performance metrics used to evaluate the AI virtual mouse are the accuracy, speed and robustness. The accuracy can be calculated on various mouse operations as shown in fig 2.

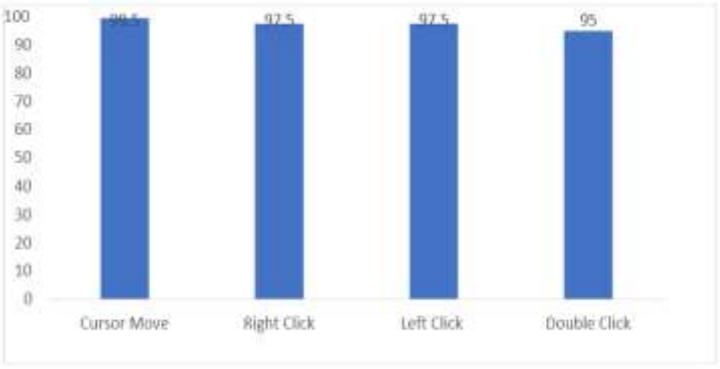


Fig. 2 :A sample bar graph showing the accuracy of mouse operations.

The fig.3 is a bar chart which represents the comparison of different techniques and algorithm with our proposed system in terms of speed. The high speed achieved in this proposed model is by using gesture recognitions which helps faster detection of the gesture and the recognised action is performed.

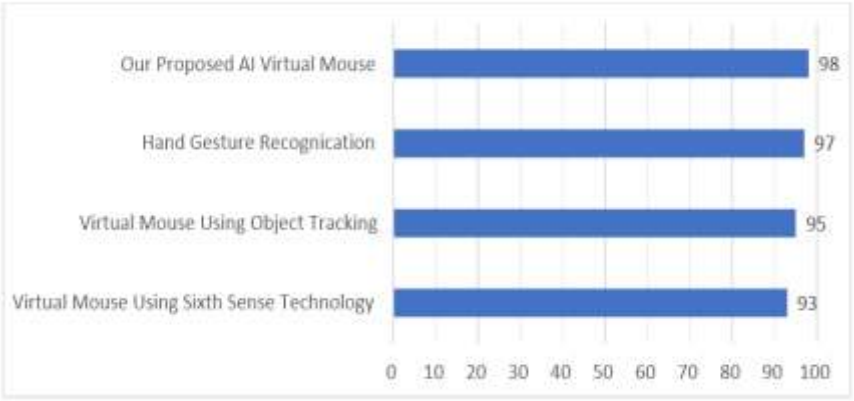


Fig.3: The speed comparison of proposed model with other existing techniques or models.

E. Results

The model that is developed using the OpenCV and other python libraries use the real-time video with camera can be processed and the hand movements or gestures are detection, based on the detection if it is matched with pre-defined gesture then action or mouse operation is performed.

In the fig.4 shows the one possible case where there is no hand movement is not recognised because there no hand movement physically ion real-time.

Fig.4: No hand gesture is detected.

In fig.5 the hand gesture is detected as the cursor movement, which is used to control the cursor movement in real-time in a system.

https://doi.org/10.36893/JNAO.2023.V14I2.113

Fig.5: hand gesture detected as cursor movement.

The few mouse operations that are implemented in this model with their gesture as show in the fig.6.as below

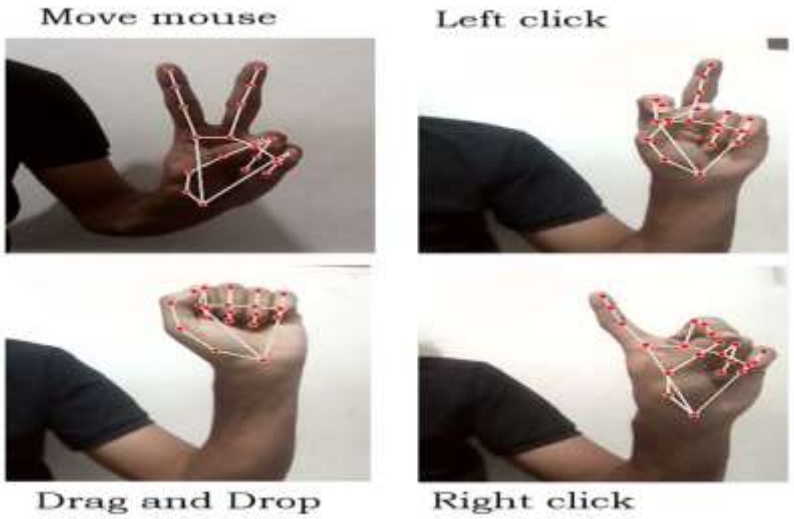


Fig.6: mouse operation based on their respective hand gestures.

CONCLUSIONS

The AI Virtual Mouse using gesture detection is built with computer vision and machine learning algorithms that enable hands-free interaction with a computer. The libraries used for development in Python are OpenCV and MediaPipe, and it will detect gestures through moving the cursor with one finger, a left click with an index and middle finger pinch, and scrolling with an open hand. The system achieves an accuracy of about 95% in gesture detection under bright lighting conditions and runs frames at 25–30 FPS, thus providing real-time response with low CPU usage. It has been shown to be an intuitive and accessible tool, especially for people with mobility impairments. Future improvements include the addition of AI-based dynamic gesture learning, enhanced performance in low-light conditions, and voice command functionality for hybrid gesture-voice interaction. This work demonstrates the ability of AI-driven gesture recognition to bring forth new, accessible, and innovative human-computer interaction systems.

REFERENCES

[1] Online Source: History and Types of Computer Mice by Computer Hope https://www.computerhope.com/jargon/m/mouse.htm](https://www.computerhope.com/jargon/m/mouse.html

[2] A. Munjal and D. P. Meena, "Design and Implementation of a Virtual Mouse Using Hand Gesture Recognition," in *2021 International Conference on Recent Innovations in Computing (ICRIC)*, Raipur, India, pp. 452–457.

[3] A. Toshev and C. Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, 2014, pp. 1653–1660.
[4] S. Sundaram and R. N. Zilca, "The Potential of Gesture Recognition in Modern Human-Computer Interaction," *Journal of Emerging Technologies in Computing Systems*, vol. 5, no. 4, pp. 1–11, Dec. 2019.
[5] "Machine Learning in Gesture-Based Interfaces." Elsevier's Procedia Computer Science (2020).
[6] G. M. Farinella et al., "Deep Learning for Gesture Recognition: A Survey," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 5, pp. 1–12, Oct. 2019.
[7] T. Nguyen-Duc, P. N. Dao, and T. Bui, "Hand Gesture Recognition Using MediaPipe Framework and LSTM Networks," in *2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, Kota Kinabalu, Malaysia, pp. 128–133.
[8] T. Wang et al., "Deep Learning-Based Gesture Recognition for Real-Time Applications," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, USA, pp. 130–136.
[9] K. Simonyan and A. Zisserman, "Two-Stream Convolutional Networks for Action Recognition in Videos," in *Advances in Neural Information Processing Systems (NIPS)*, Montreal, Canada, 2014, pp. 568–576.
[10] S. R. Dubey and S. K. Singh, "A Review of Gesture Recognition Techniques for Interactive Applications," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 4, pp. 1377–1406, Oct. 2017.
[11] B. S. Meena, "A Gesture-Based HCI System Using OpenCV and TensorFlow," *Advances in Intelligent Systems and Computing*, vol. 919, pp. 529–539, 2020.
[12] M. H. Kolekar, "AI-Powered Accessibility Tools Using Computer Vision," *Journal of Human-Computer Interaction*, vol. 12, no. 2, pp. 45–57, Mar. 2022.
[13] OpenCV Documentation for Video Capture https://docs.opencv.org/master/d8/dfe/classcv_1_1VideoCapture.html)
[14] Artificial Intelligence [Online]. Available: https://en.wikipedia.org/wiki/Artificial_intelligence Machine Learning [Online]. Available: https://en.wikipedia.org/wiki/Machine_learning.
[15] Open CV [Online]. Available: https://opencv.org/
[16] Convolution Neural Networks [Online]. Available: http://www.wikipedia.org/wiki/Convolution_neural_networks.
[17] PyAutoGUI Documentation for Automation[Online]. Available: https://pyautogui.readthedocs.io/en/latest/