

AI-Enhanced Gesture-Controlled Virtual Mouse and Keyboard

PURIMITLA BHANUCHAND*¹, MATHI PUJITHA², SAKKURTHI PUSHPALATHA³,
MOTAMARRI RAKESH⁴, YAMANA ADITHYA REDDY⁵

¹Assistant Professor, Department of CSE(AIML), Bapatla Engineering College, Bapatla 522101, AP, India

²Student, Department of CSE(AIML), Bapatla Engineering College, Bapatla 522101, AP, India

³Student, Department of CSE(AIML), Bapatla Engineering College, Bapatla 522101, AP, India

⁴Student, Department of CSE(AIML), Bapatla Engineering College, Bapatla 522101, AP, India

⁵Student, Department of CSE(AIML), Bapatla Engineering College, Bapatla 522101, AP, India

Abstract—When people want easier ways to interact with their computers, hardware inputs like keyboard and mouse are often using less. A fresh approach shows how a virtual setup can be handled by moving hands in front of any regular webcam. Instead of using on hardware devices, smart software tracks key movements using live hand gesture movements. The system turns natural gestures into the actions on screen. Ahead of its time, the online mouse lets you tap left, tap right, even scroll and also volume and brightness adjustments. By processing data in a real-time, user interactions remain smooth and responsive. From the starting, choosing Python along with OpenCV and MediaPipe makes the things clarity. Lights stays under the control and the timing stays reliable once tested. This approach is especially useful when the users are unable to perform the physical tasks using the traditional devices. In this project interacting with the hands is not required.

Key Words - Hand Gesture Recognition, Virtual Mouse and Keyboard, Computer Vision, OpenCV, MediaPipe, Human Computer Interaction, Real-Time Gesture Control

II. INTRODUCTION

A computer is operated by a mouse and keyboard. These are helpful to the people to perform the basic tasks. While using hardware input devices at every situation is not convenient. In some cases, physical interaction with hardware devices may be difficult, and not possible. So the hand gesture based interaction is simple and provides the free way of communication with the computers.

Gestures are natural movements of the human hands and can be easily performed without any special training. We can operate the computer using the hand gestures. It makes the work easier. With the development of these type of real-time hand gesture techniques, it is used to detect and track the hand movements using an ordinary camera.

The action with fingers is used to send commands to the computers. This approach allows the users to control the mouse and keyboard actions without directly touching any hardware. This project presents the AI-Enhanced Gesture-Controlled Virtual Mouse and Keyboard. The system captures the hand movements such as gestures through the camera and perform the operations such as cursor movements, clicking and typing. The main aim of this project is to provide a simple, efficient and contactless input devices. That bridges the gap between the human and computer with more flexible and user-friendly.

III. LITERATURE SURVEY

Human and computer interaction was improved with the development of artificial intelligence and computer vision. So then many researchers were tried to reduce the usage of hardware devices like mouse and keyboard and introduced the gesture based systems. These systems allow users to to control the computer using hand movements and provide the touch free interaction. In earlier research, virtual mouse systems are developed using webcams to detect hand gestures. For detecting hand gestures some methods are required like gloves to track the finger movements. Later, markerless techniques are introduced to detect hand movements directly from the camera using image processing algorithms. This made the system usage easily.

With usage of the ai models like OpenCV and MediaPipe, gesture recognition has become more accurate. Researchers are also developed the virtual keyboard using hand gestures. Even though systems work well in many cases but facing some problems like lighting and also some times gesture recognition also. In this project. An effort is made to develop a simple virtual mouse and keyboard that can works in real time and makes the computer usage easier.

IV. EXISTING SYSTEM

In the existing system, computers are operated using the physical devices like mouse and keyboard. The working of these devices is very good. But they want the physical contact. In some situations using hardware devices is not comfortable and practical also. But some researchers developed the alternative usage for the systems namely like gesture based systems. But many of those systems require the extra components like gloves, sensors, webcams to track the hand movements. This also increased the cost and complexity also. And also facing problems like lighting, background disturbances and responses delay affected the over all performance.

V. PROPOSED SYSTEM

The proposed system introduces the AI-enhanced gesture controlled virtual mouse and keyboard. It uses the web cam to capture the hand movements and processes them using the computer vision techniques like OpenCV and MediaPipe. Based on the position and movement of fingers different gestures are performed and converted into the computer commands. This system does not require any external sources like gloves and sensors. It works using only camera and software. Making it simple and cost effective. The main objective of this project is maintaining a smooth, real-time and contactless method to operate a computer while improving the interaction and accuracy.

VI. SYSTEM ARCHITECTURE

1. Input Module

- The webcam acts as an input device.
- It is used for to capture the hand movements in live video.
- The captured video is sent to the system continuously to the system.
- Clear video input is important for accurate results.

2. Pre-Processing Module

- The captured video frames are adjusted for the better quality.
- Noise and unwanted background disturbances are reduced and minimized.
- The image converted into suitable format for easy processing.
- The step improves the overall performance of the system.

3. Frame Processing Module

- The live video is divided into the individual frames.
- Each frame is analyzed separately.
- This helps in detecting hand movements clearly.
- Continuous frame processing and ensures the smooth working.

4. Hand Detection and Tracking Module

- The system detects the hand in each frame.
- The landmarks like finger tips and finger joints are identified.
- These landmarks helps the system to work more effectively.
- Accurate tracking improves the gesture recognition.

5. Gesture Recognition Module

- The finger positions are captured to identify the specific gestures.
- Based on the predefined rules, gestures are identified.
- Different gestures are used for different actions.
- For example, moving the index finger controls the cursor and some fingers combinations perform the click operations.

6. Gesture Mapping Module

- Each captured gesture is converted to a specific computer function.
- Then the system checks the whether the gesture matches the predefined conditions.
- This module ensures that correct action is performed for the gesture.

7. Command Execution Module

- Once a gesture is recognized, it is converted into the computer command.
- Then the system performs the actions according to the gesture like cursor movements, clicking, scrolling and also typing.
- This process happens continuously in real time for smooth interaction.

8. Output Module

- The final action was displayed on the computer screen.
- The cursor movements and typing appears based on the gesture performed.
- This completes the interaction between the user and the computer.

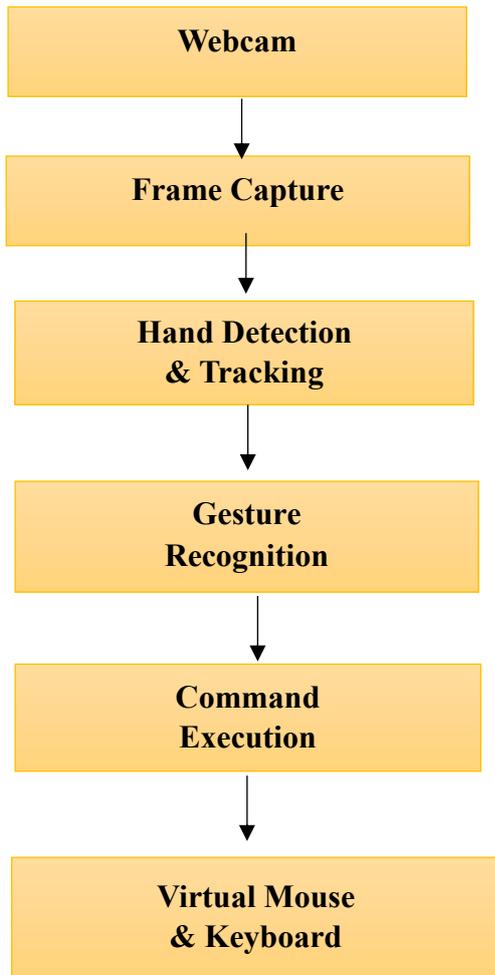


Figure1: System Architecture of AI-Enhanced Gesture Controlled Virtual Mouse and Keyboard

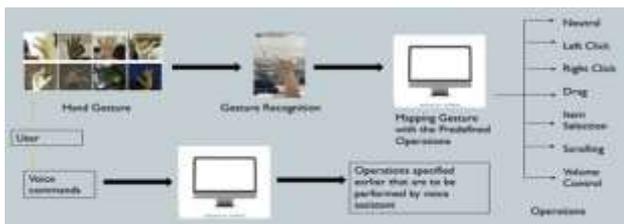


Figure 2 : System Architecture of Gesture Controlled Virtual Mouse.

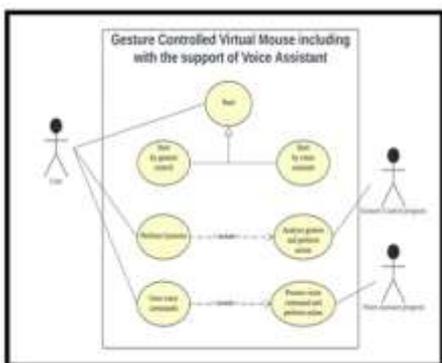


Figure 3 : Usecase Diagram

VII. METHODOLOGY

1. System Initialization

The system starts by loading the required libraries and then activate the webcam. The camera starts the capture the hand movements as input processing.

2. Video Capture and Frame Processing

The webcam captures real-time video of users hand movements. Then the captured video was divided into the several frames for easy tracking. And each frame is processed separately for best results.

3. Hand Detection and Tracking

The system detects the hand in each frame and then identifies important landmarks such as finger tips and joints of fingers. These points helps the tracking of finger movements.

4. Gesture Recognition

The position and movements of fingers are captured to identify the particular gestures. And each gesture is linked to a particular action like cursor movements, clicking, scrolling, and typing also.

5. Command Execution

After recognition the gesture, the converts the gestures into the computer commands and perform the correct actions. This process is done continuously to ensure the smooth interaction.

6. Real-Time Processing

The system performs all operations continuously without any delay. Real-time processing ensure the smooth cursor movements and quick response to gestures. So it is very useful to the users to interact with the computer without using any hardware devices.

7. Error Handling and Accuracy Improvement

The system checks the hand movements before executing the commands. It avoids the unwanted actions by confirming the gesture for a short duration. This improves accuracy and reduces mistakes during operation.

VIII. IMPLEMENTATION

The Proposed AI-enhanced gesture controlled virtual mouse and keyboard system was implemented using the Python programming language. OpenCV library was used for handling image processing tasks and MediaPipe was used for hand detection and tracking hand movements. The webcam captures the live video of the users hand movements and the video is processed frame by frame. Each frame is analyzed to detect the hand and identify important landmarks such as fingertips and finger joints. MediaPipe provides 21 hand landmark points. Which help in accurately tracking finger positions. The coordinates of the index finger are mapped to the screen resolution to control the cursor

movement smoothly. To avoid sudden jumps, smooth techniques are applied during cursor movement. The distance between the specific fingers is calculated and perform the click operations according to the hand movements. For example, when the thumb and index finger come close together then a left click operation is performed. Similarly, other finger combinations are used to performed the right click. And palm opened then it will detect the hand tracking and performed the scrolling operations and also based on another gestures other operations are also performed like brightness, volume also. A virtual keyboard layout is displayed on the screen, and based on the gestures used to select key and type characters. The system continuously captures frames, processes gestures and executes commands in real time. All modules work together inside a loop to ensure smooth and responsive interaction. The implementation was tested under different lighting conditions to check performance and accuracy. The system performs effectively in proper lighting and provides a simple and contactless way to control the computer without using physical input devices.

IX. RESULTS



Figure 1: Right click operation using AI Enhanced gesture controlled virtual mouse



Figure 2: Alphabet typing using AI Enhanced Gesture Virtual Keyboard



Figure 3: Typing using AI Gesture Controlled Virtual Keyboard

X. CONCLUSION

In this project, an AI-enhanced gesture controlled virtual mouse and keyboard system was successfully developed and implemented. The system uses webcam to detect hand movements and recognize gestures using computer vision techniques like OpenCV and MediaPipe. Based on the gestures actions like, cursor movement, clicking, scrolling, brightness and volume adjustments and typing are performed. The system works in real-time and allows users to operate the computer without physical input devices. Although in works best under lighting conditions, it still provides a simple and effective alternative for computer interaction.

XI. REFERENCES

[1] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.

[2] F. Zhang, "MediaPipe Hands: On-device Real-time Hand Tracking," Google Research, 2020.

[3] T. Nguyen and H. Nguyen, "Hand Gesture Recognition for Human-Computer Interaction Using Computer Vision," International Journal of Advanced Computer Science and Applications, 2019.

[4] A. K. Talukder and M. Hasan, "Vision-Based Virtual Mouse Control Using Hand Gesture Recognition," International Journal of Computer Applications, 2020.

[5] Python Software Foundation, "Python Language Reference," Available: <https://www.python.org>

[6] OpenCV Documentation, "Open Source Computer Vision Library," Available: <https://opencv.org>

[7] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2011.

- [8] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, 2004.
- [9] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics*, 2007.
- [10] Y. Wu and T. Huang, "Vision-Based Gesture Recognition: A Review," *International Gesture Workshop*, Springer, 1999.
- [11] S. Malik and Y. Laszlo, "Visual Touchpad: A Two-Handed Gestural Input Device," *ACM International Conference on Multimodal Interfaces*, 2004.
- [12] C. Keskin, F. Kirac, Y. Kara, and L. Akarun, "Real-Time Hand Pose Estimation Using Depth Sensors," *IEEE International Conference on Computer Vision Workshops*, 2011.
- [13] T. Starner and A. Pentland, "Real-Time American Sign Language Recognition from Video," *IEEE International Symposium on Computer Vision*, 1995.
- [14] J. Shotton et al., "Real-Time Human Pose Recognition in Parts from a Single Depth Image," *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [15] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand Gesture Recognition with 3D Convolutional Neural Networks," *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015.
- [16] M. Oikonomidis, N. Kyriazis, and A. Argyros, "Efficient Model-Based 3D Tracking of Hand Articulations," *British Machine Vision Conference*, 2011.
- [17] A. Erol, G. Bebis, M. Nicolescu, R. Boyle, and X. Twombly, "Vision-Based Hand Pose Estimation: A Review," *Computer Vision and Image Understanding*, 2007.
- [18] D. Murray and J. Little, "Using Real-Time Stereo Vision for Mobile Robot Navigation," *Autonomous Robots*, 2000.
- [19] K. Oka, Y. Sato, and H. Koike, "Real-Time Fingertip Tracking and Gesture Recognition," *IEEE Computer Graphics and Applications*, 2002.
- [20] J. Davis and M. Shah, "Visual Gesture Recognition," *Vision, Image and Signal Processing*, 199