International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 12 Issue: 12 | Dec 2025 www.irjet.net p-ISSN: 2395-0072

GESTURE CONTROL VIRTUAL MOUSE AND VOICE ASSISTANT

Prof. Arpita Patil[1], Srusti Laxani[2], Sejal Patil[3], Shifa Mulla[4], Vaishnavi Lokapur[5]

[1]*Assistant Professor, SG Balekundri Institute of Technology, Belagavi, Karnataka, India*
[2,3,4,5] *Student, SG Balekundri Institute of Technology, Belagavi, Karnataka, India*

---***---

Abstract *In this project, we combined hand-gesture control with voice commands to create a completely touch-free way of using a computer. Instead of relying on a mouse or keyboard, the system uses a camera to follow the movement of your hand. Simple actions—like moving the cursor, clicking, or scrolling—happen automatically based on how you position or move your hand in front of the camera. This makes it possible to control the computer without ever touching any device. Along with gestures, a built-in voice assistant adds another layer of convenience. By speaking naturally, users can open apps, perform searches, adjust settings, or carry out small tasks without lifting a finger. The two features complement each other well: when gestures feel easier, you can rely on your hands, and when speaking is faster, you can use your voice. During our testing, both systems performed consistently and didn't require much adjustment. The gesture tracking responded accurately, and the voice commands were understood most of the time without needing to repeat them. Because of this, the overall setup feels smooth and practical for everyday use. What makes this system especially useful is how adaptable it is. It can help people who need simpler or more accessible ways to operate a computer, such as users with limited mobility. It also works well in environments where touch-free interaction is preferred—like laboratories, healthcare settings, or situations where keeping devices clean matters. Overall, the combination of gestures and voice control creates a comfortable, flexible, and modern way to interact with technology.*

***Key Words*:** Blaze Pose, Media Pipe Framework, XG Boost, Open CV, Python Programming, Real-time Video Processing; Gesture-Controlled Virtual Mouse; Voice Assistant Integration, Touch-Free Human-Computer Interaction, Hand-Gesture Recognition.

1. INTRODUCTION

Computers play a major role in our everyday activities, but using a traditional mouse and keyboard isn't always practical—especially for people who can't use their hands comfortably or need a touch-free way to interact with a device. To solve this, our system brings together two natural forms of interaction: hand gestures and voice control. With the gesture feature, the camera follows the movement of your hands, allowing you to move the pointer, click, or select items just by shifting your fingers or palm. At the same time, the voice assistant listens for spoken commands, making it possible to open applications, search the internet, adjust settings, or perform routine tasks simply by talking. By combining both features, the system offers a smooth, intuitive, and completely hands-free way of using a computer. This makes everyday interactions easier, reduces physical effort, and supports users who need simpler or more accessible alternatives to traditional input devices. It also works well in environments where touching equipment isn't ideal, such as laboratories, hospitals, or public spaces. Overall, the blended use of gestures and voice commands creates a more natural and flexible computing experience.

2. LITERATURE REVIEW

People who know a lot about computers have found new ways to use them without a mouse or keyboard. By using hand movements, users can control the pointer and select items just by moving their hands. In addition, speaking to the computer allows users to give commands using their voice. By combining these two methods makes computer interaction easier and hands-free. This interaction these techniques are highly effective for improving usability, enabling multitasking, and working in environments where physical contact is not possible. In the paper "Hand Gesture Recognition-Based Virtual Mouse Using Media Pipe" by Pavithra et al., the authors introduced un touch-free way to control computer by tracking hand movements through a webcam. Using media pipe, the system identifies a finger positions and converts them into action like moving cursor or clicking. While the approach is simple and effective, especially for basic computer interactions it is performance drops in poor lighting, and offers only limited gesture options. The system also does not include other helpful input modes—such as voice control—which would make it more accessible in different environments.

1. In this paper "Virtual Mouse is using Hand Gestures" by Matlani et al., developed a system that allows the user to operate the mouse through basic hand gestures movements captured through webcam. Using OpenCV and Media Pipe, the system detects fingertips to perform tasks such as controlling the cursor, clicking, dragging. It offers a touch free way to interact with a computer, but its performance drops in poor lighting or busy backgrounds, and it relies only a gesture without additional input options.

2. In the paper "Virtual Mouse Control Using Colored Fingertips and Hand Gesture Recognition" by Reddy et al. Developed a system that controls the mouse using either colored fingertips caps or bare-hand gesture detected through OpenCV. Their method performs action like cursor movements and clicking by identifying colors, contours, and convex in the real time. 3.
3. In the paper "Gesture and Voice Controlled Virtual Mouse for Elderly People" by Krishna Dharavath et al., a system was developed to assist elderly individuals in controlling computer using both gesture and voice commands. The System tracks hand movements through camera for cursor control and recognizes voice commands for action like clicking or scrolling. By integrating these two methods, the system offers an accessible and hands- free way for elderly user to interact with technology, overcoming challenges like limited dexterity.
4. In the study "Gesture Controlled Virtual Mouse with Voice Assistant Integration" by Jain et al., the writers made a setup that allows people to use a computer through hand motions and spoken instructions. The idea is that you can guide the cursor and click on items simply by waving your hand, and verbal commands can be used to open software or carry out various tasks. This combination of moving your hand and speaking commands gives those who have limited or no hand movement an easier method for computer use, offering an uncomplicated, hands-free method to operate and manage equipment.
5. In the paper "Gesture and Voice Controlled Virtual Mouse: A Review" by Kumar et al., the authors take a deep dive into different ways we can control a computer using gestures and voice commands instead of a traditional mouse or key board. They check out different tools that allow people to control the pointer using hand motions and select items or do things by saying instructions. This way of doing things is really good for individuals who find it hard to use normal ways of putting information in, such those who cannot move or handle things easily.
6. The writing talks about how putting hand movements and talking together makes these setups simpler for people to use and get to, but it also brings up some problems that these tools still have.

3. PROPOSED SYSTEM

3.1 Overview

The main goal of the work is to design a no-contact interaction system that brings together hand-gesture recognition with voice command processing. It makes use of a webcam for real-time tracking and identification of hand motions to perform basic performing computer tasks without touching any input devices. Simultaneously, a voice assistant module will capture spoken commands using a microphone and interpret speech based on various speech- recognition methods. These two modules will put together an intuitive and accessible interface that improves usability for everyday users and supports individuals with mobility limitations. The machine-learning model is integrated with real-time landmark detection and efficient audio processing in this project to ensure fast, accurate, and user-friendly performance.

3.2 HAND LANDMARK MODEL

The hand landmark model is designed to recognize and track, in real time, key points on the human hand. It detects key joints such as the fingertips, knuckles, and wrist from input either from a webcam or an image, creating a skeletal representation of the hand. This mapping enables the model to interpret different hand shapes, movements, and gestures with great accuracy. This information is then used as input for gesture-recognition algorithms, enabling applications such as virtual controls, sign-language interpretation, and hands-free human–machine interaction. The lightweight architecture in this system supports fast processing and its possible use in real-time scenarios across a many different platforms

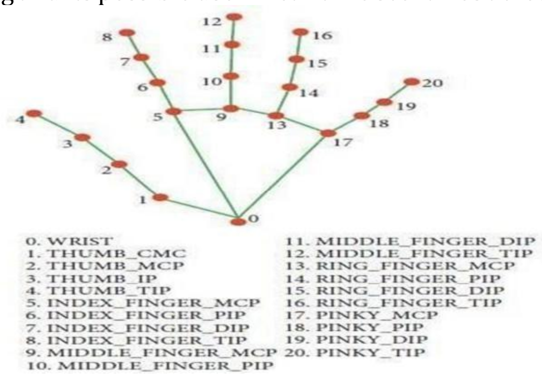

Fig 3.1: Hand Landmark Model

3.3 Voice assistant system model

The proposed model of a voice assistant system processes the user's speech in several key stages: it starts with the stage of speech acquisition, where the user's voice is captured, filtered, and prepared for analysis. After that, the speech recognition layer turns speech into text that is interpreted by the NLU module to determine intent and context. The dialogue management system then selects an appropriate response, and the NLG module renders a natural, text- like response. This response is then converted into speech by the Text-to-Speech engine. Backend integration provides the link to online services and user data, and a feedback loop allows learning and continuous improvement.

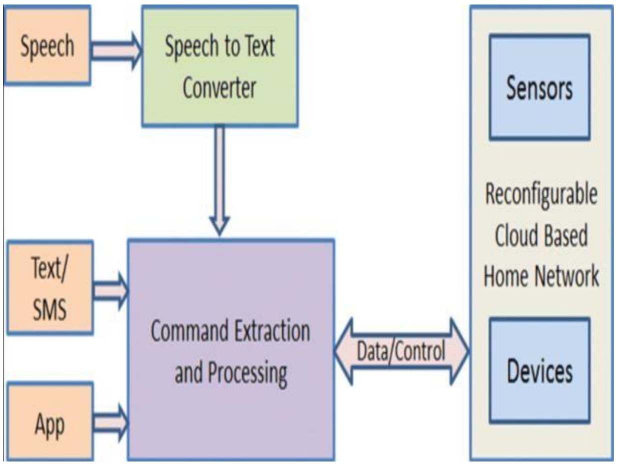

Fig 3.2 : Voice Assistant Model

4. METHODOLOGY

The system proposed for the AI-based virtual mouse utilizes the laptop or PC camera for live video capturing. The OpenCV library in Python activates the webcam and continuously records frames. The system then evaluates these frames with the help of a trained AI model that identifies the movement of the user's hand. The system uses Media Pipe to identify the user's dominant or non-dominant hand. It can recognize up to two hands at a time, provided that the confidence of detection and tracking is above 50%. The condition of each finger is decided by checking its position, returning 1 if the finger is open and 0 if it is closed. This is done by calculating the distance ratios between the fingertip and the middle and base knuckles. For example, for the index to pinky fingers, it uses the landmark points [8, 5, 0], [12, 9, 0], [16, 13, 0], and [20, 17, 0], respectively, while for the thumb it uses [6].

Pointer on the screen when both the index and middle fingers are raised. Each hand is described by 21 landmarks, with x and y coordinates normalized between 0.0 and 1.0, and a z-value indicating the hand's proximity to the camera. These correspond to 3D positions relative to the center of the hand. The system also estimates whether a hand is left or right with a probability greater than 50% for that classification.

Gesture recognition is used to control various mouse functions, such as:
Pointer movement: A V-shaped gesture, raising the index and middle fingers, moves the mouse pointer across the x- and y-axes.

Drag-and-drop/Multi-selection: A fist gesture activates these functions.
Brightness/Volume control: A pinch with the dominant hand controls horizontal brightness and vertical volume.

1. Scrolling: A pinch with the non-dominant hand activates horizontal or vertical scrolling.
2. 2. Left click: Performed by extending only the middle finger.
3. Right click: Performed by extending only the pointer finger
4. Double click: Performed by closing both the two fingers next to the thumb.
5. All mouse actions are performed through the Py Auto GUI module, which sends commands directly to the operating system to perform standard mouse operations.

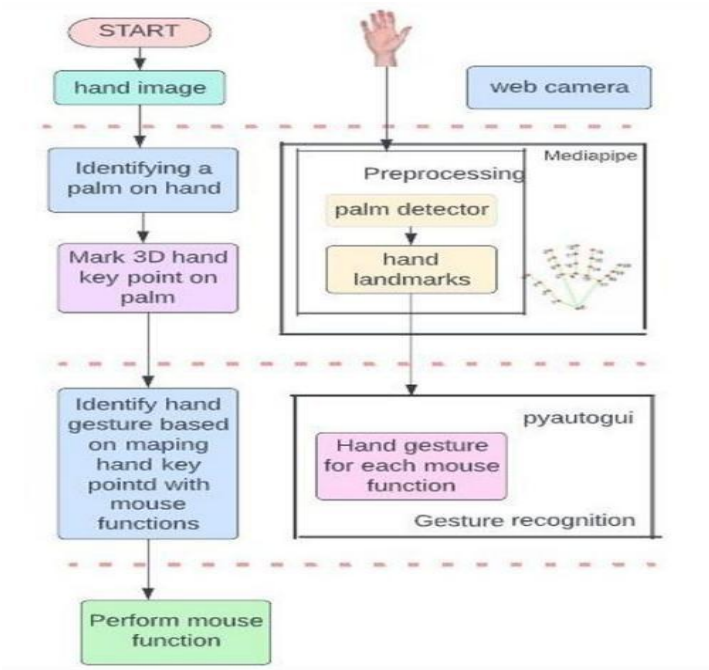

Fig 4.1: Flowchart of gesture control

This voice-assistant module is developed through a Media Pipe enables the system to move the mouse sequence of stages that will ensure accurate Speech interaction. First, the system captures the user's voice through a microphone and removes background noise through audio preprocessing techniques. The cleaned audio will then be given to an automated system that converts speech into text for. This text is then analyzed by a natural language understanding component to identify the user's intent and relevant keywords. Based on these inputs, the dialogue management unit selects the most suitable response or action. Finally, a text-to- speech engine converts the generated response back into speech, thus allowing natural and continuous communication between the system and the user.

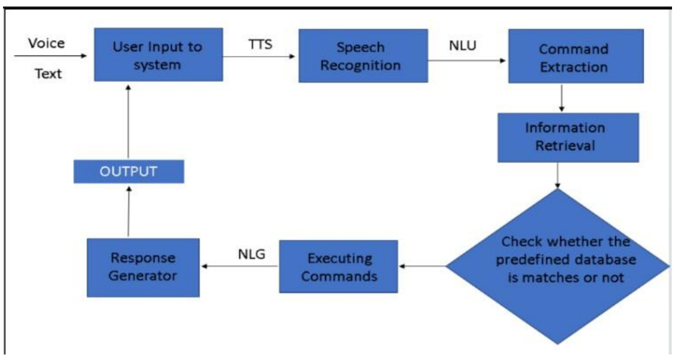

Fig 4.2: Flowchart of voice assistant

4.1 Block Diagram

The diagram illustrates the working architecture of a control system that integrates voice and gesture commands to operate robotic components. The system begins with two types of input — voice commands and gesture commands — which are both transmitted to the Adafruit IO platform. Adafruit IO acts as a communication interface, processing and sending the received data to the ESP32 module. The ESP32 module functions as the main controller, interpreting the incoming commands and executing the corresponding actions. Based on these instructions, the ESP32 controls various robotic components, including DC motors and servo motors, to perform specific robotic movements. This setup enables real-time control of robotic systems using both voice and gesture inputs, offering a flexible and user-friendly interaction method.

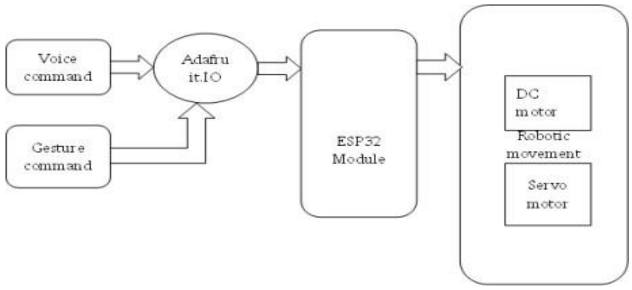

Fig 4.3: Block diagram of proposed system

5. ALOGORITHM

1. Input Capture:

- Read video frames from the camera.
- Listen to audio from the microphone.

2. Hand Processing:

- Detect the hand and fingertip points.
- Check which fingers are raised?
- Smooth the movement by mapping the tip of the index finger to the coordinates on the screen.
- Recognize the following gestures: two-finger slide (scroll), pinch (click), and pinch-move (drag).

3. Voice Processing:

- Translate speech into text when it is detected.
- Recognize basic commands such as "click," "scroll down," "open browser," etc.

4. Decision Layer:

- Execute a mouse event if a gesture corresponds to an action.
- When a voice command is detected, carry out the appropriate action.
- When both occur simultaneously, voice commands take precedence over gestures.

5. Output:

- Depending on the detected gesture or spoken command, move the cursor, click, drag, scroll, or open applications.

6. ADVANTAGE

- This system makes using a computer feel more natural because you can control it without touching anything.
- You are able to shift your hand to perform actions or simply speak when it's easier than typing.
- This helps a lot in situations where your hands are occupied, dirty, or you're not sitting close to the device.
- It also supports people who are not comfortable with typical methods input tools by giving them an easier way to interact. Since it only needs a normal webcam and a microphone, it doesn't require expensive equipment and can run on most computers.
- By combining gestures with voice commands, the system creates a smoother, quicker, and more comfortable way to operate a computer in everyday tasks.

7. APPLICATIONS

- This system can be used in many real-life situations where hands-free control is helpful.
- It is useful in smart homes, where users can operate lights, music, or appliances using voice or gestures. In workplaces, it helps people control presentations or switch between files without touching the computer, which is especially convenient during meetings.
- It can also be helpful for people who have physical disabilities by giving them an easier way to interact with technology. In medical environments, doctors can use gestures or voice commands to view reports without touching devices, keeping things hygienic. The system is also effective for gaming, virtual learning, and controlling multimedia, making interactions faster and more interactive.
- Overall, it is suitable anywhere hands-free, touch- free, or more natural computer control is needed.

8. RESULTS

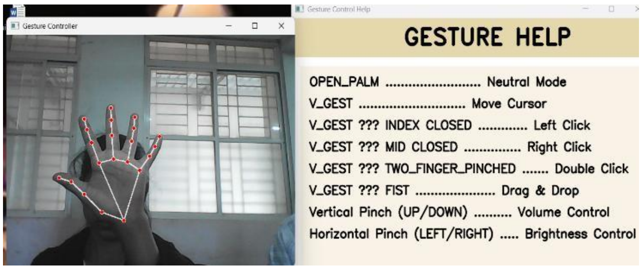

Fig 8.1: neutral mode

If all five fingers remain open — with the tip IDs 0, 1, 2, 3, and 4 detected — the system understands that the user does not want to perform any action. In this state, shown in Fig-8.1, the computer simply stays idle and no mouse movement or click is triggered on the screen.

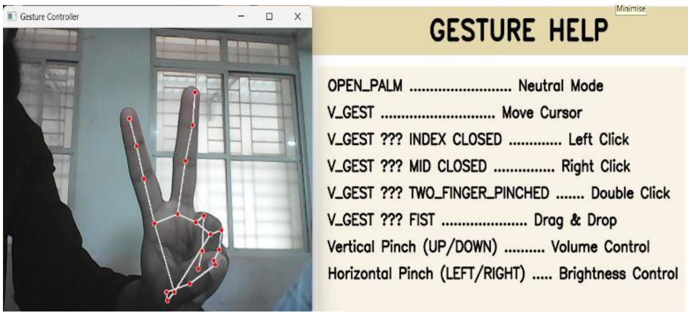

Fig 8.2: Move cursor

This feature allows you to move the mouse pointer around the computer screen. When both the index finger (tip ID = 1) and the middle finger (tip ID = 2) are raised, the Python Auto Py package moves the cursor accordingly, as illustrated in Fig-8.2

Fig 8.3: Drag and Drop

.

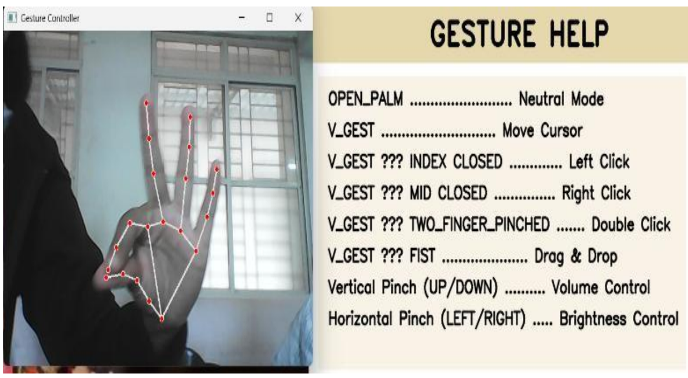

Fig 8.4: volume control

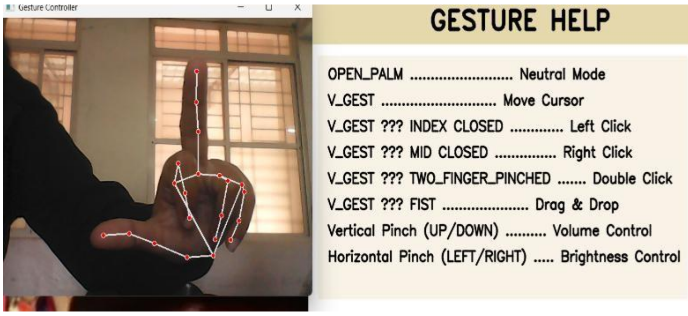

Fig 8.5: Left click

To perform the left click we need to close the index finger

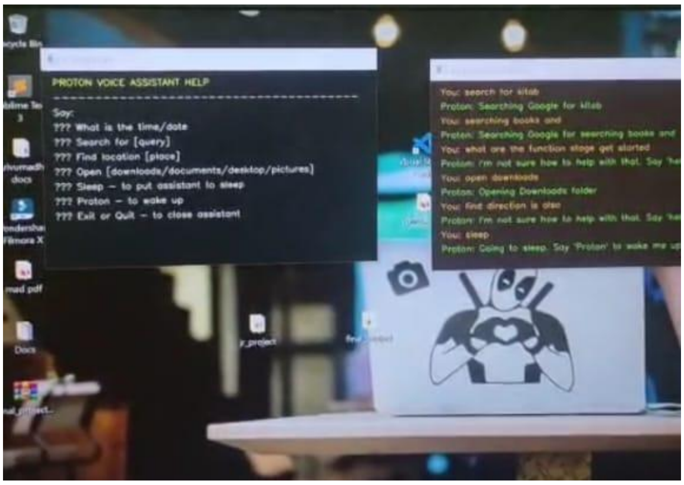

Fig 8.6: Voice Assistant

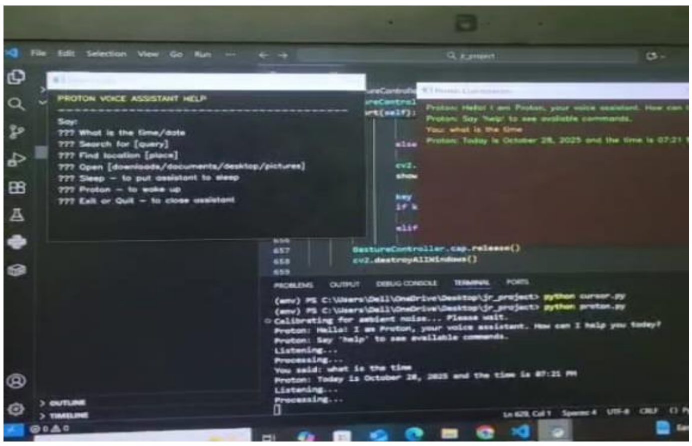

Fig 8.7: Proton Voice Assistant

9. CONCLUSION

The gesture-control and voice-assistant system provides a simple, natural, and touch-free way of interacting with a computer. By combining hand-tracking with speech recognition, The system enables users to perform this function effectively and comfortably without relying on traditional input devices. It works smoothly in real time, requires only basic hardware, and can be used in many practical situations such as smart homes, presentations, and accessibility support. Overall, the system shows how AI-based interaction can make computers more intuitive, user-friendly, and accessible for users from various backgrounds.

10. REFERENCE

[1] N. R. Sathish Kumar, A. Venkateswara Reddy, B. Anil Kumar, B. Pavan Kalyan Reddy and A. Shashank Kumar Reddy, "Hand Gesture-Based Virtual Mouse with Advanced Controls Using OpenCV and Mediapipe," Proceedings of theInternational Conference on Power, Energy, Control and Transmission Systems (ICPECTS), IEEE, 2024.

[2] Matlani, R., Mishra, S., Dadlani, R., Tewari, A., & Dumbre, S., "Virtual Mouse using Hand Gestures," 2021 International Conference on Technological Advancements and Innovations (ICTAI), IEEE, pp. 340–345, 2021.

[3] VishnuTeja Reddy, V., Dhyanchand, T., Vamsi Krishna, G., & Maheshwaram, S., "Virtual Mouse Control Using Colored Finger Tips and Hand Gesture Recognition," National Institute of Technology Warangal, pp. 1–8.

[4] Akash Singh, Shraddha Sagar, Shreya Bhatt, & Trapti Upadhyay, "Real Time Virtual Mouse System Using Hand Tracking Module Based on Artificial Intelligence," 2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI), IEEE, pp. 36–39, 2023

[5] AI Based Virtual Mouse with Hand Gesture and AI Voice Assistant Using Computer Vision and Neural Networks — J. Kumara Swamy, Navya V. K. (2023).

[6] Kavyasree K., Sai Shivani K., Sai Balaji T., Sahana Yele & Vishal B., "Hand Glide: Gesture-Controlled Virtual Mouse with Voice Assistant," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 12, issue IV, April 2024, pp. –, DOI: 10.22214/ijraset.2024.61178.