World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(RESEARCH ARTICLE)

Check for updates

# Wave Talk: A smart gesture and voice assistant

Parwateeswar Gollapalli, Sana Tabasum *, Sidhartha Tadaboina, Sai Kumar Ganta and Aishwarya Gottipamula

*Department of CSE (Data Science), ACE Engineering College, Telangana, India.*

World Journal of Advanced Engineering Technology and Sciences, 2025, 15(02), 073-081

Publication history: Received on 18 March 2025; revised on 29 April 2025; accepted on 01 May 2025

Article DOI: https://doi.org/10.30574/wjaets.2025.15.2.0512

## Abstract

Wave Talk is a multimodal human-computer interaction system that integrates real-time hand gesture recognition and voice command processing to enable seamless, touchless control of digital devices. Utilizing OpenCV and Media Pipe for gesture tracking, alongside Speech Recognition and pyttsx3 for voice interaction, the system offers an intuitive interface accessible to users across diverse environments, including those with physical disabilities or in hygiene-sensitive settings. Designed to run on standard webcams and microphones, Wave Talk ensures cost-effectiveness and broad usability. The methodology encompasses data acquisition, preprocessing, model integration, and action execution, with system testing confirming high accuracy and low latency. Applicable in smart homes, healthcare, education, and public spaces, Wave Talk demonstrates the potential of multimodal interaction systems to enhance accessibility, efficiency, and user experience in next-generation smart technologies.

**Keywords:** Gesture Recognition; Voice Assistant; Multimodal Interface; Media Pipe; OpenCV; Speech Recognition; Touchless Control; Real-Time Interaction

## 1. Introduction

In the modern digital landscape, there is an increasing demand for more intuitive, natural, and inclusive ways for humans to interact with machines. Traditional input devices such as keyboards, mice, and touchscreens, while effective, are limited in flexibility and often require physical effort or contact, making them unsuitable for users with mobility impairments or in environments where hygiene is a concern. As technology becomes more integrated into daily life—through smart homes, healthcare systems, and public service kiosks—there is a pressing need for alternative control methods that align more closely with human behavior and comfort.

Gesture recognition and voice-based interaction have emerged as promising alternatives that cater to the need for touchless, hands-free communication with digital devices. Gesture interfaces allow users to perform actions through simple hand movements, offering a natural mode of control without physical contact. Similarly, voice assistants enable command execution through speech, enhancing ease of use and efficiency. However, systems that rely solely on either modality may not be sufficient in all scenarios—noisy environments may hinder voice recognition, while gesture detection may be limited by lighting conditions or hand visibility.

WaveTalk introduces a hands-free interaction system that integrates gesture and voice recognition to control digital devices naturally and intuitively. It aims to improve accessibility and user experience in environments where traditional input methods are limited or impractical.

WaveTalk is designed to bridge this gap by integrating both gesture and voice recognition into a single multimodal interface. By combining the strengths of both input methods, WaveTalk enables more adaptable and context-aware interaction. The system utilizes the MediaPipe framework and OpenCV for real-time hand gesture detection, alongside

* Corresponding author: Sana Tabasum

speech recognition technologies and text-to-speech engines to facilitate responsive voice communication. This dual-input model not only improves interaction reliability but also increases accessibility for users with varied needs and environmental constraints.

The motivation behind WaveTalk stems from the goal of enhancing user experience by making technology more responsive, inclusive, and human-centric. The project targets applications in domains where touchless interaction is critical, such as healthcare, assistive technologies, smart homes, and educational tools. With its lightweight implementation and reliance on standard hardware like webcams and microphones, WaveTalk offers a scalable, cost-effective solution for the next generation of intelligent systems. Through this research, the project aims to contribute to the broader field of human-computer interaction by promoting multimodal systems that align more closely with human communication patterns.

### 1.1. Problem Statement

The conventional dependence on physical input devices and single-mode interfaces presents several critical limitations, especially for users in accessibility-sensitive environments or contexts demanding hands-free operation:

- **Physical Dependency and Inaccessibility:** Traditional input methods such as keyboards, mice, and touchscreens require physical contact and manual dexterity. This poses challenges for individuals with physical disabilities, elderly users, or those recovering from injuries, making technology less inclusive and usable (Kumar et al., 2023; Hassan et al., 2024).
- **Lack of Flexibility in Varied Environments:** Voice-only systems often fail in noisy surroundings or public spaces where speech recognition becomes unreliable. Likewise, gesture-only systems can be limited by lighting conditions, occlusions, or hardware sensitivity, making them unsuitable as a standalone solution (Patel et al., 2023; Singh & Raj, 2024).
- **Limited Multimodal Integration:** Most interaction systems are designed around a single input type—either touch, voice, or gesture. This lack of integration limits user adaptability and results in reduced efficiency, especially in dynamic contexts like smart homes, healthcare, or education where different modes may be more appropriate at different times (Nair et al., 2023; Bhargav et al., 2024).

To address these challenges, there is a growing need for an intelligent, adaptive, and inclusive interface that combines multiple input methods—particularly gesture and voice—in a single, seamless system. Such a solution must be real-time, lightweight, cost-effective, and compatible with common hardware like webcams and microphones.

### 1.2. Objectives

This research aims to explore the development and implementation of a multimodal human-computer interaction system that integrates both voice and gesture recognition for seamless, real-time, and touchless device control. The specific objectives of this study include:

- To review existing human-computer interaction methods, including traditional and assistive technologies, highlighting their usability challenges and accessibility limitations.
- To investigate the application of computer vision and speech processing techniques in creating intuitive and inclusive interaction systems.
- To identify the technical and user-experience gaps in single-mode systems, such as voice-only or gesture-only interfaces, particularly in accessibility-sensitive or hands-free environments.
- To design and develop a conceptual and functional framework for WaveTalk—a unified system that combines hand gesture recognition and voice command processing using MediaPipe, OpenCV, and SpeechRecognition technologies.
- To evaluate the system's performance in terms of recognition accuracy, response time (latency), and user adaptability across varied use cases.
- To outline the potential applications, limitations, and future research directions for enhancing multimodal interaction systems in fields such as healthcare, education, smart homes, and assistive technologies.
- By achieving these objectives, the paper seeks to contribute to the evolution of accessible, intelligent interfaces that can redefine human-computer interaction and broaden the usability of technology for diverse populations and environments.

## 2. Literature review

### 2.1. Existing Methods

The domains of gesture recognition and voice-controlled systems have evolved significantly with the advancement of machine learning and sensor technologies. Numerous standalone solutions have been developed, each tailored to specific applications such as entertainment, assistive technology, and smart environments. However, most of these systems operate in isolation, either focusing solely on gesture recognition or voice interaction, lacking seamless integration.

#### *2.1.1. Gesture Recognition Technologies*

Early gesture recognition systems primarily relied on sensor-based devices. Tools such as data gloves and inertial measurement units (IMUs) captured finger and hand movements with high precision. Although accurate, these methods are not user-friendly due to the requirement of wearing specialized hardware.

In contrast, vision-based systems like Microsoft Kinect introduced markerless motion tracking using RGB-D cameras. While Kinect enabled touchless control and gained popularity in gaming and education, it required dedicated hardware and controlled lighting conditions to function effectively.

Another notable system, Leap Motion, tracks hand and finger movements in three dimensions using infrared sensors. It offers precise tracking but has limitations in field of view and sensitivity to ambient lighting, making it less effective in dynamic environments.

#### *2.1.2. Voice Assistant Technologies*

Voice assistants such as Google Assistant, Amazon Alexa, and Apple Siri have become widely adopted in both consumer and enterprise applications. These systems rely on natural language understanding and speech-to-text models to interpret user commands. Despite their success, these assistants often underperform in noisy settings or when users have speech variances, such as accents or speech impairments.

For offline and privacy-focused applications, open-source tools like CMU Sphinx and Vosk provide alternatives that function without constant internet connectivity. However, these tools typically require fine-tuning and offer lower recognition accuracy compared to commercial models.

#### *2.1.3. Multimodal Interaction Systems*

Recent research has explored multimodal interfaces that combine gesture and voice inputs for enhanced usability. Some experimental systems integrate convolutional neural networks (CNNs) for visual gesture recognition and recurrent neural networks (RNNs) or transformers for audio signal processing. While these prototypes show promise in providing more natural interaction, they often suffer from synchronization issues, computational complexity, or a lack of scalability for real-world deployment.

Systems such as Intel RealSense and Google Soli represent efforts to create compact, hardware-integrated gesture sensing solutions. Intel RealSense uses depth-sensing cameras for gesture and face tracking, while Soli leverages radar technology for fine motion detection. Despite their innovation, they are not yet accessible for wide-scale implementation due to hardware dependencies and cost.

### 2.2. Proposed Method

To overcome the limitations observed in existing single-mode systems, WaveTalk introduces a comprehensive solution that combines gesture recognition and voice command processing into a single, unified framework. The objective is to deliver a seamless, touchless interface for interacting with digital devices, particularly beneficial in environments where physical contact is impractical or for users with accessibility needs.

#### *2.2.1. System Overview*

WaveTalk is built upon two primary input streams:

- **Hand Gestures**, tracked in real-time using **MediaPipe's hand tracking framework**, which detects 21 key points on the human hand.

- **Voice Commands**, interpreted using the **SpeechRecognition** library, which leverages Google's speech-to-text API to convert spoken instructions into executable actions.

These two modalities operate in tandem, allowing users to perform actions such as adjusting volume, scrolling through documents, launching applications, or controlling multimedia using either gestures, voice, or both.

### 2.3. Core Components

*2.3.1. Gesture Recognition Module:*

- Utilizes webcam input to detect hand presence and position.
- Processes landmark data to identify predefined gestures (e.g., palm, pinch, fist).
- Maps gestures to system-level functions such as mouse movement, clicking, scrolling, or brightness/volume adjustment.

*2.3.2. Voice Recognition Module*

- Continuously listens for voice input via microphone.
- Converts speech to text and matches it to a command set (e.g., "open notepad", "play music").
- Executes matched actions using pyautogui for automation.

*2.3.3. Multimodal Synchronization*

- The system dynamically prioritizes input streams based on confidence levels and user context.
- Allows simultaneous gesture and voice use without interference, enabling fluid user interaction.

*2.3.4. Control Execution Layer*

- Uses **PyAutoGUI** to simulate mouse and keyboard actions based on input triggers.
- Implements real-time feedback (visual/audio) to confirm user actions.

## 3. Methodology

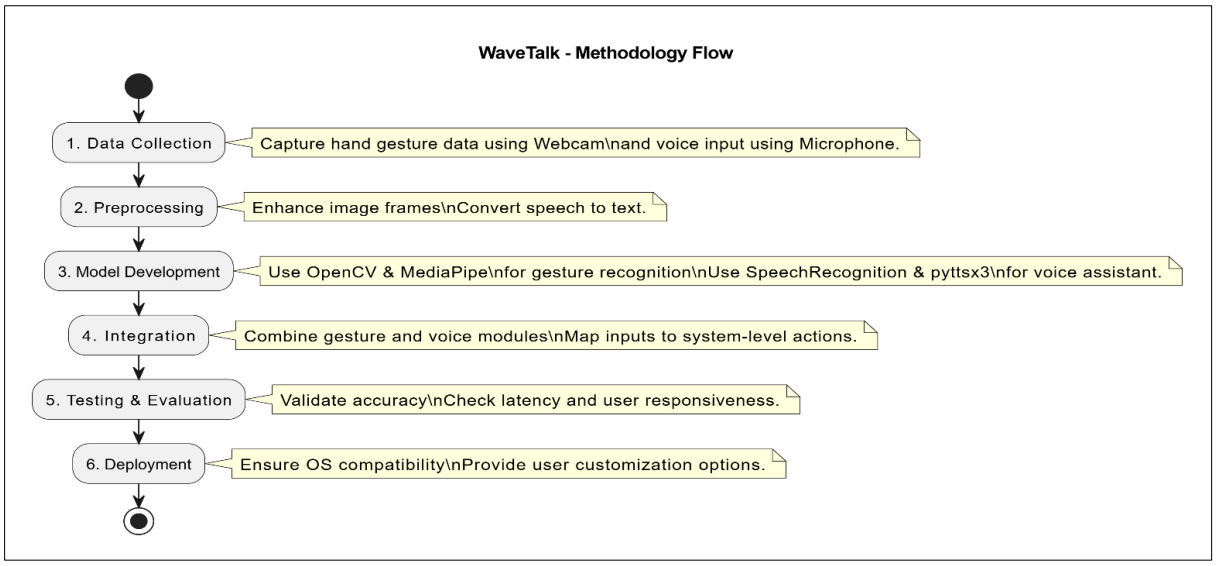


**Figure 1** Methodoloogy

The methodology of **WaveTalk** follows a systematic approach to enable real-time gesture and voice-based interaction:

*3.1.1. Input Acquisition*

- **Webcam** captures live video for hand gesture tracking.
- **Microphone** records voice commands.

*3.1.2. Preprocessing*

- For gestures: Image frames are processed using **OpenCV** and landmarks are extracted using **MediaPipe**.
- For voice: Audio is filtered for clarity before transcription using **SpeechRecognition**.

*3.1.3. Recognition and Mapping*

- Gestures are classified based on finger positions and hand movement patterns.
- Voice input is converted to text and matched to predefined commands.

*3.1.4. Action Execution*

- Recognized inputs are mapped to corresponding system actions (e.g., click, scroll, open apps) using **PyAutoGUI**.

*3.1.5. Feedback and Adjustment*

- Visual or audio cues confirm successful actions.
- Thresholds and mappings are fine-tuned through real-time testing for improved accuracy.

This approach ensures a smooth, real-time, and touch-free user experience that can adapt to various environments and user needs.

### 3.2. System Architecture

The architecture of **WaveTalk** is designed to integrate gesture and voice inputs into a unified control system, enabling hands-free interaction. It consists of the following key components:

*3.2.1. System Components*

Input Devices

The system uses a webcam to capture real-time hand gestures and a microphone to record voice commands for processing. These input devices serve as the primary channels through which users interact with WaveTalk, enabling seamless gesture tracking and speech recognition.

Gesture Recognition Module

The system employs OpenCV in combination with MediaPipe to accurately detect and track hand landmarks in real time. By recognizing specific hand gestures, it enables seamless interaction with the computer system, allowing the user to control the mouse cursor, perform click actions, and scroll through content. This gesture-based approach enhances accessibility and offers a touch-free method of navigation.

Voice Recognition Module

The system incorporates the SpeechRecognition library to convert spoken words into text, enabling voice-based interaction. Recognized voice commands are then mapped to specific system actions, such as launching applications, adjusting the volume, or performing other predefined tasks. This voice control functionality adds convenience and hands-free accessibility to the overall user experience.

Preprocessing Unit

By enhancing image quality and filtering background noise in voice input, the application ensures more precise recognition and interpretation. Gesture data is also normalized to support consistent and accurate tracking throughout the interaction.

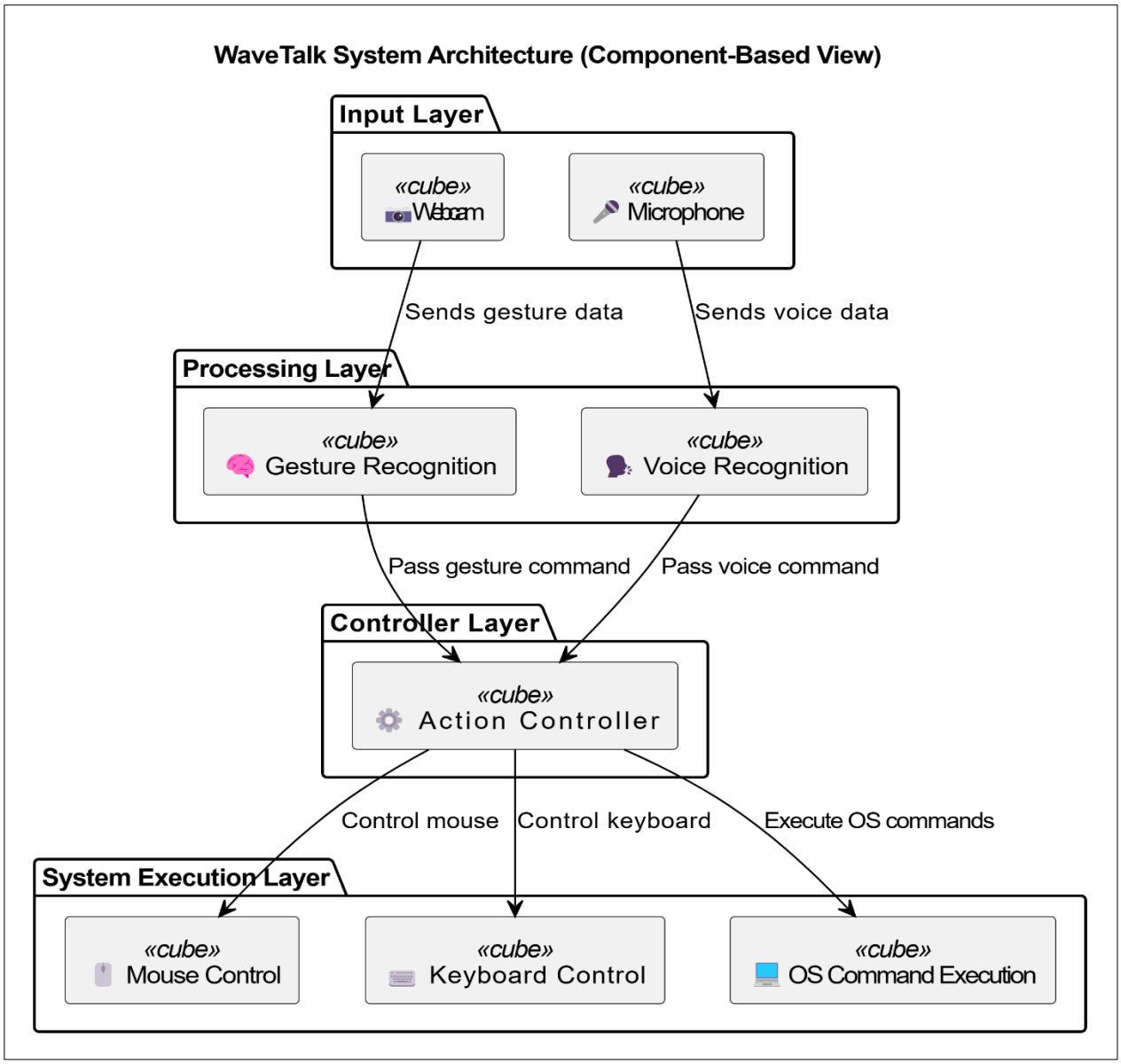


**Figure 2** System Architecture

### 3.3. Command Mapping & Execution

Detected gestures and voice commands are mapped to specific operating system functions, enabling intuitive control over the system. PyAutoGUI is employed to simulate mouse movements and keyboard inputs, allowing the application to execute actions like clicking, typing, or navigating the interface seamlessly.

#### *3.3.1. User Interface (UI) & Feedback System*

Real-time visual feedback is provided to indicate detected gestures and recognized voice commands, enhancing user awareness and interaction. Additionally, the application offers customization options, allowing users to adjust gesture sensitivity and configure voice command settings to suit their preferences and environment.

#### *3.3.2. Error Handling & Optimization Module*

The system actively monitors misclassifications in both gestures and voice commands, offering users the flexibility to retrain specific gestures or redefine voice inputs for improved accuracy. It also focuses on optimizing response time and recognition precision to ensure a smooth and responsive user experience.

This modular and scalable architecture ensures that WaveTalk remains adaptable for future enhancements, including integration with smart devices and emerging technologies.

## 4. Results and Discussion

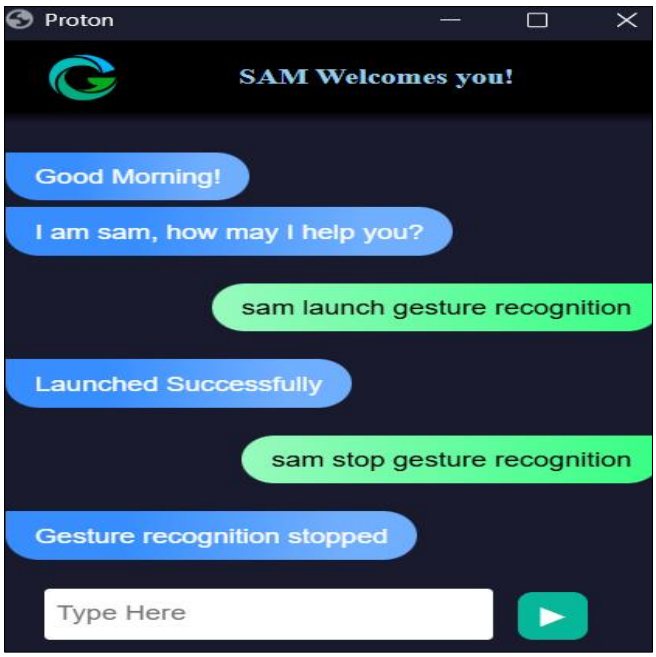


**Figure 3** Voice Assistant

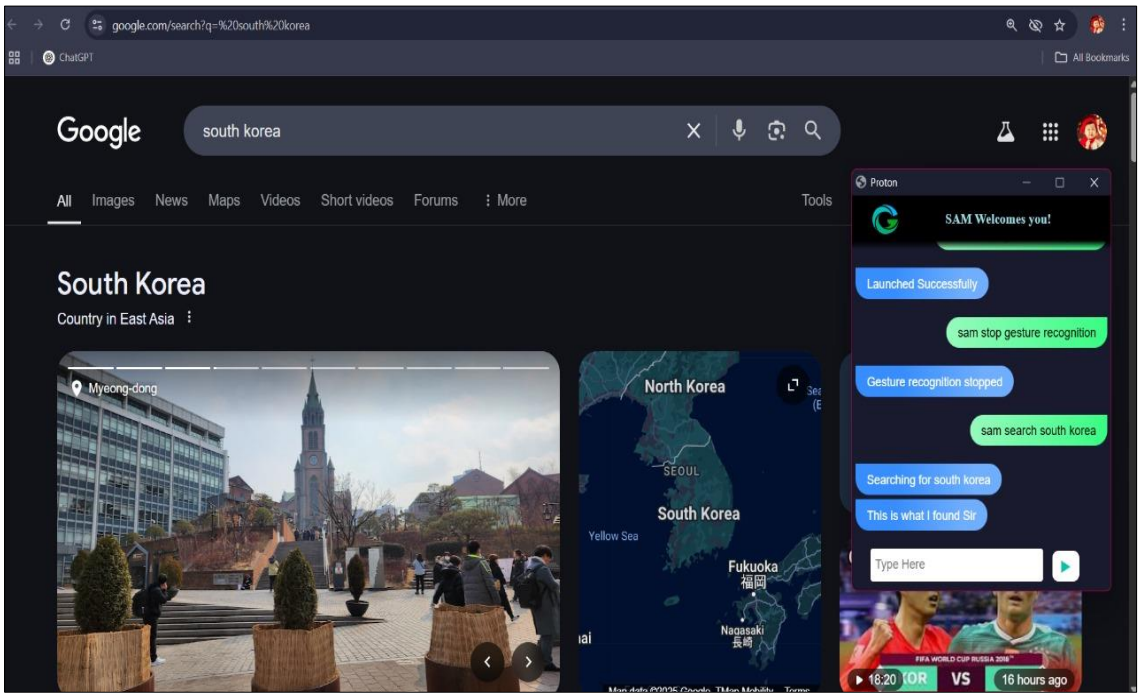


**Figure 4** Voice Assistant – Search

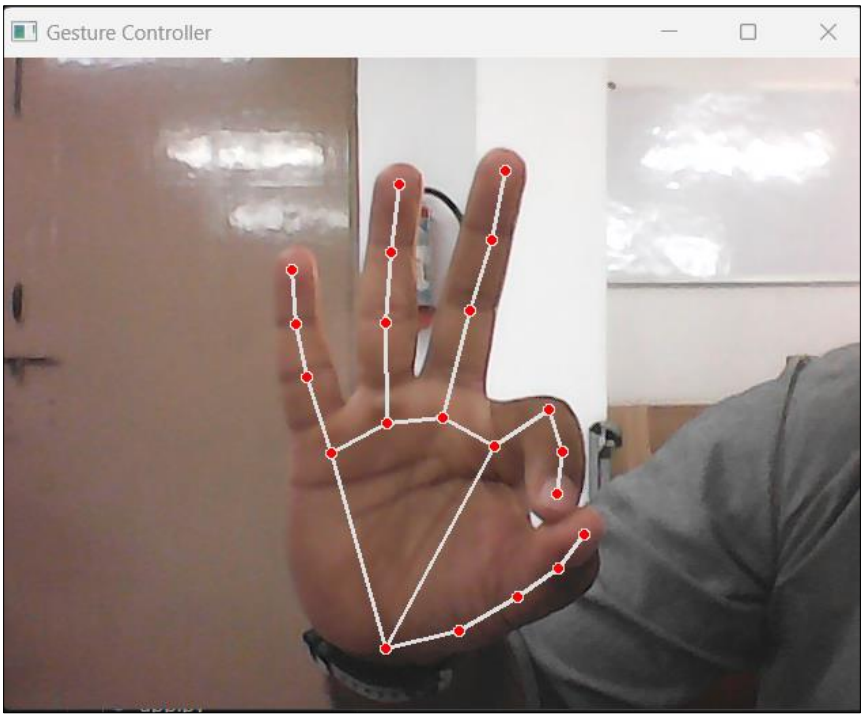


**Figure 5** Smart Gesture – Volume Operator

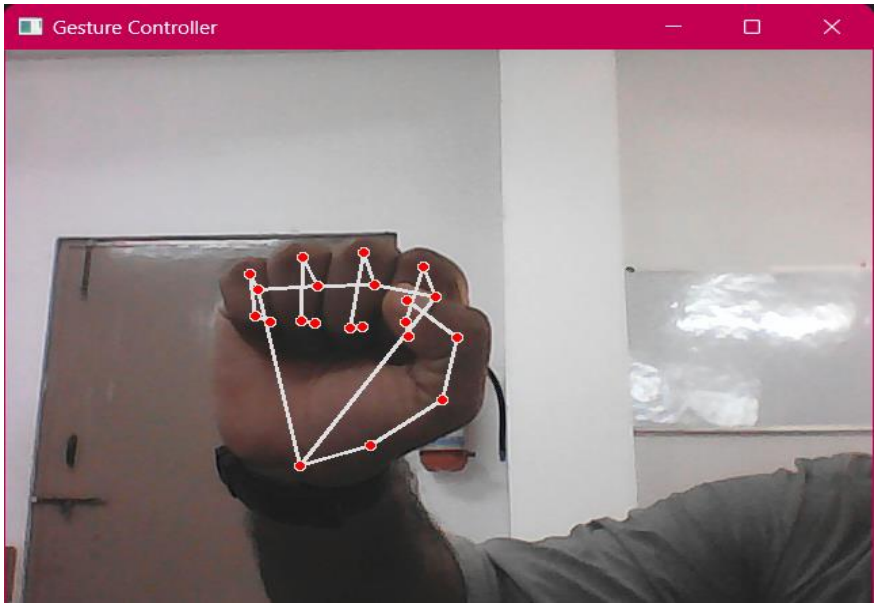


**Figure 6** Smart Gesture – Drag and Drop

## 5. Conclusion

WaveTalk presents an innovative approach to human-computer interaction by seamlessly integrating gesture recognition and voice control into a single, touchless interface. By leveraging open-source tools like MediaPipe and SpeechRecognition, the system provides real-time, accessible, and intuitive control over digital devices without the need for specialized hardware. Its dual-mode input design enhances usability in a variety of environments, particularly benefiting users with mobility impairments or in hands-free scenarios. With its modular architecture and scalability, WaveTalk lays a strong foundation for future advancements in assistive technology, smart automation, and immersive user experiences.

## Compliance with ethical standards

*Disclosure of conflict of interest*

There is no conflict of interest.

## References

[1] C. J. Cohen, G. Beach and G. Foulk, "A basic hand gesture control system for PC applications", Proceedings 30th Applied Imagery Pattern Recognition Workshop (AIPR 2001). Analysis and Understanding of Time Varying Imagery, pp. 74-79, 2001.

[2] R. Runwal et al., "Hand Gesture Control of Computer Features", Lecture Notes in Mechanical Engineering, 2021, [online]

[3] S.S. Abhilash, L. Thomas, N. Wilson and C. Chaithanya, "Virtual Mouse Using Hand Gesture", International Research Journal of Engineering and Technology (IRJET), vol. 5, no. 4, pp. 3903-3906, 2018.

[4] E. Erdem, E. Yardimci, Y. Atalay, and V. Cetin, Computer vision-based mouse, Acoustics, Speech, And Signal Processing, Proceedings. (ICASS). IEEE International Conference.

[5] Hojoon Park, A Method for Controlling the Mouse Movement using a Real-Time Camera, Brown University, Providence, RI, USA, Department of computer science.

[6] F. S. Khan, M. N. H. Mohd, D. M. Soomro, S. Bagchi, and M. D. Khan, 3D hand gestures segmentation and optimized classification using deep learning, IEEE Access 9,131614–131624 (2021).

[7] OpenCV.org. (n.d). Open Source Computer Vision Library

[8] Banerjee, A., Ghosh, A., Bharadwaj, K., & Saikia, H. (2014). Mouse control using a web based colour detection .arXiv preprint arXiv:1402.4722.

[9] A. M. Patil1, S. U. Dudhane1, M. B. Gandhi, " Cursor Control System Using Hand Gesture Recognition", International journal of advanced research in computer and communication engineering, vol. 2, issue: 5, May 2013.