

Enhancing Interactive Presentation Systems Using AI-Based Vision Gesture Recognition

Aditya Raosaheb Wagh¹, Jayesh Gajanan Chaudhari², Moin Mukhtar Bagwan³, prof. Ashwini Triloksing Devarale⁴

¹UG Student, Dept. of Computer Engineering, KCE's College of Engineering and Management, Jalgaon, India

²UG Student, Dept. of Computer Engineering, KCE's College of Engineering and Management, Jalgaon, India

³UG Student, Dept. of Computer Engineering, KCE's College of Engineering and Management, Jalgaon, India

⁴Assistant Professor, Dept. Of Computer Engineering, KCE's College of Engineering and Management, Jalgaon,India

Abstract –

Traditional classrooms use control devices such as keyboards, mouse, and laser pointers. These devices limit interaction and flexibility for the presenter. This research paper presents an AI-based vision gesture recognition system focused on improving presentation control and maintaining the presenter's flow. The proposed system uses computer vision to detect hand gestures, allowing users to control slides and perform drawing actions without any physical contact with devices.

The system is developed using computer vision techniques and trained gesture patterns to ensure smooth and accurate interaction. The proposed approach enhances reliability, reduces dependency on external devices, and provides a more natural presentation experience.

Key Words: Gesture Recognition, Computer Vision, AI-Based Interaction, Interactive Presentation, Vision-Based Control

1.INTRODUCTION

Traditional presentation systems are limited because they depend on basic devices like mouse, keyboard, and laser pointer. These devices are not flexible, and they often disturb the flow of the presenter. As a result, viewers may lose interest and get bored. Nowadays, attention span is shrinking, and long presentations make the audience tired and less engaged.

Presenters face problems such as breaking their flow while switching slides or handling devices. Using a keyboard or mouse requires physical interaction and

sometimes even a separate space or person to manage it. This can reduce the presenter's focus and also distract the audience, which breaks the overall presentation flow. Gesture-based control is better because it does not require any physical device to control the presentation. Hand gestures are used directly to change slides and perform actions. This makes the presentation more natural and smooth. It helps maintain the flow of the presenter and keeps the audience more engaged compared to traditional presentation methods.

2. Proposed System

2.1 System Overview

The main objective of this project is to make the presentation process smooth and avoid breaking the flow of the presenter. This system is suitable for presentations in education, commercial meetings, seminars, and other professional environments.

In traditional presentations, physical devices like keyboard or mouse are required to control slides. This may reduce audience attention and sometimes the presenter needs another person to navigate the PowerPoint, which can disturb both the presenter and the audience.

The system is simple to use. The user uploads a presentation file in formats such as .ppt, .pptx, or .pdf. The recently uploaded four files are displayed in the system dashboard. The user selects and opens the required file in the web application. Once opened, the file pages are displayed, and the user can navigate the slides using hand gestures. The camera view is also visible, and the user can adjust the display size.

The system works in real-time. After uploading the file, slide navigation happens instantly using hand gestures.

This system is developed specifically from a presentation point of view, where the presenter uploads the file and controls it using hand movements without physical devices.



Figure 1: System Overview

2.2 Working Principle

The system currently runs on localhost as a MERN stack-based web application. When the project runs, the dashboard opens where recently uploaded files are displayed. There is also an option to upload a new file and start presenting it.

The camera captures the hand gestures. The system detects only hand gestures, mainly palm and finger movements for slide navigation. Using OpenCV and MediaPipe, the hand landmarks are tracked and converted into numeric coordinate values. These numeric values are processed by the system logic to identify specific gestures and perform slide navigation. If the palm is not detected, the system remains idle. Once the palm is detected, the pointer appears and the system starts recognizing gestures for slide navigation.

2.3 Hardware Requirements

- Laptop or PC (minimum 8GB RAM recommended)
- Webcam (built-in or external camera)
- Stable lighting environment
- Internet connection (only for initial setup if required)

2.4 Software Requirements

- Operating System: Windows / Linux
- Programming Language: Python, JavaScript
- Backend Framework: Node.js
- Frontend Framework: React.js
- Database: MongoDB
- Computer Vision Libraries: OpenCV
- Hand Tracking Library: MediaPipe
- Web Browser: Google Chrome or any modern browser

3. System Architecture

3.1 Overall Architecture Diagram

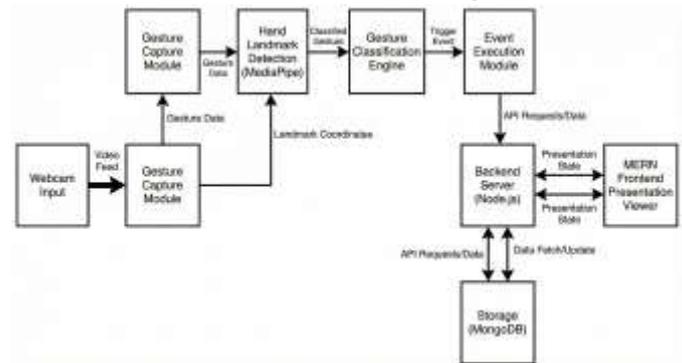


Figure 2: System Architecture

There several models main models are gesture capture module and event execution module in between that hand landmark detected by MediaPipe and gesture is classified and event is triggered.

The data flow is kind of simple like webcam input then it fed the video to gesture capture module this gesture capture module give gesture data to hand landmark detection which used MediaPipe to locate hand landmark this gesture is classified using gesture classification engine and event is triggered then event getting in execution module that call the api in backend server node.js and then it implemented in the frontend and user see the event where triggering.

3.2 Processing Module

- **Gesture capture module** – it taking the input from webcam and feed the video data then this gesture data is submitted to hand landmark detection module.
- **Gesture Classification Engine** – it take classified gesture and trigger event to event execution module.
- **Event execution Module** – it take the trigger event and api requires to backend server for triggering that event from particular gesture.

4. Methodology

The methodology of the proposed system is based on real-time gesture detection and event execution. The system is divided into multiple modules that work together to ensure smooth slide navigation.

First, the webcam continuously captures live video frames. These frames are sent to the Gesture Capture Module, which processes the input video stream. Each frame is analyzed to detect the presence of a hand.

The captured frame is then passed to the Hand Landmark Detection module, which uses MediaPipe to identify key hand landmarks. MediaPipe detects multiple coordinate points on the hand such as

fingertips, joints, and palm center. These landmarks are converted into numeric coordinate values.

After extracting the landmark data, the Gesture Classification Engine processes these values. The system compares the landmark positions with predefined gesture patterns. Based on this comparison, a specific gesture such as “Next Slide” or “Previous Slide” is identified.

Once the gesture is classified, an event is triggered. This event is passed to the Event Execution Module, which sends an API request to the backend server built using Node.js. The backend processes the request and updates the frontend accordingly. The user can then see the slide navigation happening in real time.

If no hand is detected, the system remains idle and no event is triggered. This helps reduce false commands and improves reliability.

5. Performance Evaluation

To evaluate the performance of the system, several tests were conducted under normal indoor lighting conditions. The system was tested using multiple gesture attempts to measure accuracy and response time.

The gesture recognition accuracy was calculated by comparing the number of correctly detected gestures to the total number of gesture attempts. The system achieved an average accuracy of approximately **93%** during testing.

Response time was measured as the time difference between performing a gesture and observing the slide change on the screen. The average response time was approximately **200 milliseconds**, which is suitable for real-time presentation use.

The system was tested at different distances from the camera to observe detection stability. It performed best within a range of approximately **0.5 to 1.5 meters** from the camera.

Lighting conditions were also observed. In well-lit environments, gesture detection was stable. However, performance slightly decreased in low-light conditions.

6. Results and Discussion

The results show that the proposed system successfully enables real-time gesture-based slide navigation. The gesture recognition mechanism works efficiently under normal indoor lighting and standard presentation setups. The system maintains smooth interaction without requiring physical devices like a keyboard or mouse. This improves presentation flow and allows the presenter to move freely while controlling slides.

One of the major strengths of the system is its simplicity and cost-effectiveness, as it only requires a webcam and a standard laptop. The integration with a web-based platform also makes it easy to use.

However, certain limitations were observed. The system performance may reduce under poor lighting conditions or if the hand is partially blocked. Background complexity may also slightly affect detection accuracy. Additionally, very fast hand movements may sometimes cause minor delays or misclassification.

Overall, the system demonstrates that AI-based vision gesture recognition can effectively enhance presentation interaction and reduce dependency on traditional control devices. With further improvements such as IoT integration and advanced gesture optimization, the system can become more robust and widely usable in smart classroom environments.

3. CONCLUSIONS

The proposed system successfully demonstrates a real-time AI-based vision gesture recognition system for interactive presentations. By detecting hand gestures, presenters can control slides, perform drawing actions and navigate content without relying on physical devices like a keyboard, mouse, or remote. This improves the overall flow of the presentation, keeps the audience engaged, and makes the presentation process smoother and more natural.

The system is simple, cost-effective, and works efficiently in normal indoor lighting conditions. Performance evaluation shows high gesture recognition accuracy and low response time, making it suitable for real classroom and professional environments. Overall, this work proves that gesture-based interaction can be a practical and effective alternative to traditional presentation methods.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my college faculty and project guide for their support and guidance throughout this project. I am also thankful to my peers and team members who helped during development and testing. Special thanks to the developers of OpenCV and MediaPipe libraries, whose tools made real-time hand gesture recognition possible.

REFERENCES

1. R. S. Kori, P. C. K. Reddy, and P. Hiremath, "Real Time Hand Gesture Recognition for Human Computer Interaction", *International Journal of Computer Applications*, Vol. 180, Issue 3, pp. 1–6, 2018.
2. M. Zhang and Y. Tian, "Fingertip Detection for Human Computer Interaction Using Deep Learning", *IEEE International Conference on Image Processing*, pp. 156–160, 2019.
3. S. Molchanov, X. Yang, and E. Learned Miller, "Online Detection and Classification of Dynamic Hand Gestures", *ECCV Conference*, pp. 219–234, 2016.
4. H. Liang and J. Yuan, "Fast and Robust Hand Pose Estimation Using Deep Learning", *IEEE Transactions on Image Processing*, Vol. 28, Issue 10, pp. 4806–4816, 2019