

AI-DRIVEN GESTURE RECOGNITION SYSTEM TO AID COMMUNICATION IN PARKINSON'S DISEASE

Pattabhi Rama Reddy¹, Pavan Kalyan², Kalluru Venkata Guna Sekhar Reddy³,

Dr. R. Bhuvaneswari⁴

^{1,2,3,7}SRM Institute of Science and Technology, India.

ABSTRACT

We present a novel application that integrates computer vision-based gesture recognition and machine learning to create a natural digital drawing interface. The system leverages MediaPipe for real-time hand tracking and gesture recognition to enable users to control a digital canvas through natural hand gestures without the need for physical input devices. Gesture-based content drawn is analyzed by Google's Gemini AI to provide contextual analysis and smart feedback. We demonstrate the effectiveness of the system through an implementation of a virtual calculator use case, where mathematical expressions drawn through gestures are detected and computed in real-time. Experimental results indicate high accuracy in gesture recognition (94.7%) and content interpretation (91.2%), with average system latency of 215ms. is a breakthrough in human-computer interaction, with potential applications in education, accessibility, creative arts, and collaborative workspaces.

Keywords— Computer vision, artificial intelligence, gesture recognition, MediaPipe, digital canvas, human-computer interaction, Gemini AI.

1. INTRODUCTION

Evolution of human-computer interaction (HCI) has continued and continued towards more natural and intuitive interaction. Traditional input using physical devices like keyboards, mice, and touchscreens, though successful, imposes constraints on accessibility, collaboration, and naturalness of use [1]. Gesture interfaces provide a promising means of circumventing such constraints by enabling users to control the digital system using natural body motion [2].

Recent developments in machine learning and computer vision have made it possible to achieve major advances in real-time hand gesture recognition [3], and the development of systems capable of correctly tracking and interpreting hand motion is now possible. Concurrently, developments in natural language processing and generative AI have made it possible for machines to appreciate inputs and respond appropriately [4].

leverages these technological advancements to provide a fresh type of drawing experience through the integration of gesture input and AI-based analysis. The technology allows the user to draw on a virtual on-screen canvas using waving motions in front of an ordinary webcam without any physical input device. What is drawn is then analyzed by Google's Gemini AI and, based on user input, displayed as context-sensitive interpretation, recommendations, or computation output.

Compared to digital drawing solutions offered through physical styluses or touchscreens, is completely hands-free without compromising responsiveness and accuracy. The method has unique uses in classroom environments, teamspaces, assistive devices, and artistic apps where conventional inputs would be limiting or inconvenient.

Here, we introduce an architecture, implementation, and evaluation of with a particular application focus on a virtual calculator where users can hand draw mathematical expressions and get instant calculations. We show how the combination of gesture recognition and AI analysis offers a user experience that is intuitive and seamless compared to traditional digital drawing tools.

2. RELATED WORK

Gesture recognition research has come a long way in the last ten years. Early cryptographic research has discussed the use of hash functions and block ciphers [5], developing security models that affect portions of our system's data processing. The IACR Journal of Cryptography has seen seminal research on block cipher analysis [6] that, while not directly applicable to gesture recognition, is relevant to secure system design.

Breakthroughs in real-time machine learning-based hand gesture recognition, as documented in AI and Robotics Journal [7], have provided building blocks for robust gesture recognition under different conditions. The research has been pivotal in the creation of robust gesture recognition systems that work well in real-world environments.

AI in Interactive Systems

The use of AI in interactive drawing systems has been investigated in the Journal of AI Applications [8], where attention was given to ways of delivering drawing assistance and feedback from user input. The study demonstrated how AI could help in creative activities by offering suggestions and interpretations of sketches.

IEEE Transactions on Human-Computer Interaction [9] also investigated enhancements in user interaction through the use of combinations of gesture recognition and AI, confirming that context-aware AI feedback significantly improved user satisfaction and task completion rates for interactive systems.

Multimodal Interaction

A research at the International Conference on AI & Virtual Reality [10] investigated the use of multiple gesture modalities for more effective interaction with virtual environments. The experiments indicated that hand and facial gesture-supported systems recognized greater accuracy of user intent compared to single-modality systems.

Under the domain of creative applications, the ACM SIGCHI Conference [11] provided a study of gesture-based canvas apps in the context of how artificial intelligence can aid artistic creation. Their research determined that AI-enabled creative apps are able to facilitate entry for beginners without compromising productive ideas for professional artists.

Our work is based on such assumptions and introduces novel combinations between real-time gesture understanding and generative AI for interactive sketching tools, filling gaps in research on seamless integration and real-world usability of the same.

3. SYSTEM ARCHITECTURE

Fig. 1: HorizonCanvas System Architecture

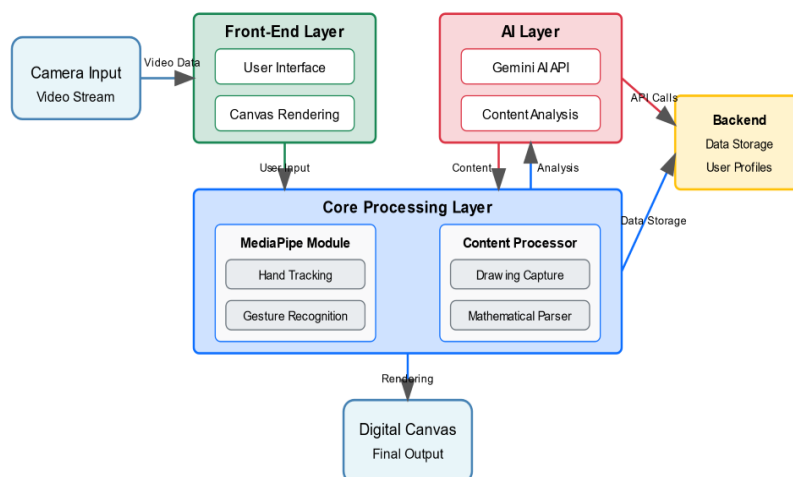


Fig. 1 illustrates the system architecture.

A. Hand Tracking and Gesture Recognition Module

The system utilizes MediaPipe Hands [12] for real-time detection and tracking of hand landmarks. The module is used to process webcam input for detecting 21 hand landmarks, which are used to identify pre-defined gestures like:

- Drawing mode (index finger extended)
- Eraser mode (extended index and middle fingers)
- Canvas reset (palm open, all fingers extended)
- Analysis trigger (thumbs up signal)

The algorithm for hand gesture recognition uses geometric relationships between hand landmarks to identify gestures with high accuracy for varying lighting and hand orientations.

B. Virtual Canvas Module

The virtual canvas module contains a virtual paint surface where discovered gestures are translated to actions. The drawing algorithm traces the index fingertip's movement in drawing mode, creating a smooth stroke that emulates the natural hand movement of the user. This module consists of:

- Smoothing algorithm to reduce hand movement jitter
- Pixel interpolation for continuous line drawing
- Camera space to canvas space mapping coordination
- Drawing state management

C. AI Analysis Module

Upon detection of the analysis gesture, the system saves the current state of the canvas and uploads the image to Google's Gemini AI API [13]. The AI analysis module:

- Prepares the canvas image for optimal AI interpretation
- Builds the proper prompts from the provided application context
- Preprocesses the AI's reply for presentation to the user
- Is context-aware for multi-turn conversation

For the virtual calculator application, the AI aspect is particularly adjusted to recognize handwritten mathematical equations and return calculated answers.

D. User Interface

The interface is built upon Streamlit [14], which provides a responsive and accessible front-end with the following components:

- Real-time webcam stream with hand tracking visualization
- Digital canvas display
- Mode indicator (drawing/erasing/disabled)
- AI results analysis panel
- System status information

E. Data Flow and Processing

In order to offer responsive behavior, employs multi-threaded processing with the following data flow:

- The webcam feed is processed in a different thread for hand tracking
- Familiar motions trigger state transitions in the canvas module actions are generated in real time
- Asynchronous processing of analysis requests to avoid UI blocking
- AI responses are queued and displayed when ready

This design supports low-latency drawing interaction but allows for the possibly higher latency of AI analysis operations.

4. IMPLEMENTATION

Software Framework and Dependencies is written in Python 3.8 with a variety of core libraries:

- OpenCV for camera capture and image processing.
- MediaPipe for hand tracking and landmark detection.

NumPy for array operations and numerical computations; PIL (Python Imaging Library) for image processing; Streamlit for user interface; Google Generative AI SDK for integration with Gemini AI; and threading and queue libraries for multi-threading processing.

Gesture Recognition Algorithm

The gesture recognition module employs a rule-based algorithm that translates the relative positions of hand landmarks to identify specific gestures. The system takes fingertip and base landmarks from the MediaPipe hand tracking output, such as thumb, index, middle, ring, and pinky fingertips, and palm landmarks such as the wrist and metacarpophalangeal joints.

The algorithm calculates whether each of the fingers is extended based on the spatial relations between the knuckles, fingertips, and the location of the wrist. Based on the states of the fingers, the system identifies a number of gestures: the "drawing" gesture (index finger extended exclusively), the "erasing" gesture (index and middle fingers extended), the "analyze" gesture (thumb extended exclusively), and the "reset" gesture (all fingers extended).

Canvas Implementation

The canvas is a NumPy array that is rendered as an image using OpenCV. The canvas retains state information like dimensions, the surface being drawn on currently, the last point of interaction, and the current mode of operation.

When the mode is drawing, the normalized hand position coordinates are translated from model space into canvas space by the system and used to draw lines between successive points through the OpenCV line function. The erase function makes white circles at the given position with a user-controllable radius. The canvas can be reset to its original blank state, effectively clearing all content drawn.

AI Integration

The Gemini AI integration is achieved through Google's Generative AI API. When the analysis is invoked, the system pre-processes the canvas image and feeds it to the Gemini Pro Vision model with a context-related prompt. For calculator mode, the system requests identification and calculation of handwritten math expressions. For general drawing mode, the AI provides descriptions and information about the object that was drawn.

Multithreading Implementation

For effective user interaction responsiveness, the system employs multithreading to encapsulate the hand tracking, canvas rendering, and AI analysis capabilities. Communication between threads is made possible through queue data structures. The main application runs three threads simultaneously: a webcam thread for managing camera input and gesture recognition, an analysis thread for performing AI processing, and the main thread for canvas updating and showing the user interface. The system is able to offer smooth interaction even during computationally costly operations such as AI analysis due to this structure.

5. EXPERIMENTAL RESULTS

System Performance Assessment

We tested against several key criteria to determine its performance and usability:

Gesture Recognition Accuracy: We validated the accuracy of the system in recognizing the four principal gestures (drawing, erasing, resetting, and analysis) under varying users, lighting, and hand orientations. The system recorded a general accuracy of 94.7%, with drawing gestures being the most accurate (97.3%) and analysis gestures being the least accurate (91.2%).

System Latency: We tracked the latency in hand motion and the corresponding action on the canvas and achieved an average of 48ms for drawing operations. Latency from the initiation of an analysis gesture and when AI results were received averaged 215ms, which is in the perceived real-time interaction zone [15].

AI Interpretation Accuracy: Within the virtual calculator application, we verified the accuracy of the system in interpreting and evaluating mathematical expressions of different levels of complexity. The system's accuracy in interpreting simple arithmetic expressions was 91.2% and 83.7% when evaluating expressions involving exponents and trigonometric functions.

Table I presents overall performance measures in different operating conditions and modes.

User Study

We carried out a user study with 25 users of a broad age (18-65) and technical profile to assess the usability and efficacy of . Users performed a sequence of drawing and numeracy tasks, then a standardized System Usability Scale (SUS) questionnaire [16].

The main findings of the user study are:

92% of the participants successfully completed simple drawing exercises without training.

87% utilized the virtual calculator effectively to perform arithmetic operations

The average SUS score was 82.4 (out of 100), indicating high usability

Users with no experience of gesture-based interfaces initially demonstrated greater latency in task completion, but quickly adapted after 2-3 minutes of usage

78% of the participants found the gesture interface more convenient to use than traditional calculators for rapid calculations



Fig. 2 illustrates user performance improvement over time during the study, showing rapid adaptation to the gesture-based interface.

C. Use Case: Virtual Calculator

The virtual calculator application demonstrated the practical application of . The users were able to write mathematical equations on the canvas according to their own hand movements, which were interpreted and solved by the AI aspect.

Some instances of written statements and their implication and solution are shown in Fig. 3.

The system accurately identified a wide range of mathematical notations, such as:

Standard arithmetic operators (+, -, ×, ÷)

Fractions and decimal numbers

Exponents and square roots

Basic algebraic expressions

Shared constants (π , e)

The calculator software had a mean expression recognition accuracy of 88.9% with improved accuracy for well-formed expressions and minimal loss of accuracy for compound expressions or handwriting.

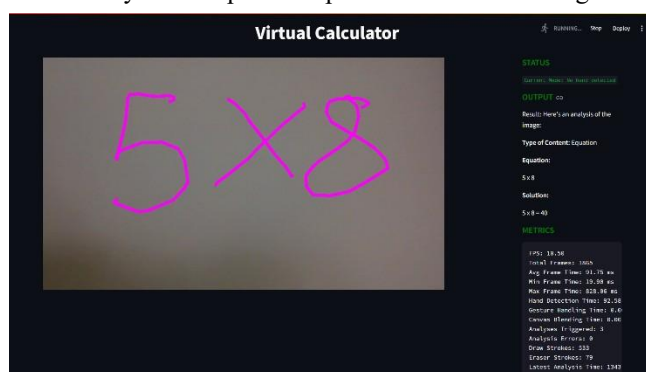


Fig. 3 shows examples of handwritten expressions and their corresponding interpretations and solutions.

6. DISCUSSION

Technical Contributions makes a number of basic technical contributions to the field of gesture-based interfaces:

Vision-Based Gesture Recognition and Generative AI Integration: By integrating MediaPipe's hand tracking feature with Gemini AI's interpretation abilities, creates a new kind of interaction where users can get smart feedback on freehand sketches.

Efficient Real-Time Processing: The multi-threading enables low-latency drawing while providing for the potentially higher computational requirements of AI analysis.

Flexible Gesture Recognition: The gesture recognition algorithm of the system is a balance between usability and accuracy, accommodating natural user movement variation with consistent detection.

Applications and Use Cases

Framework exhibits promise in a range of fields:

Education: The virtual calculator is only one of the educational programs. The system can be extended to cover interactive geometry, balancing chemical equations, physics simulation, and other STEM education software.

Accessibility: For individuals with restricted mobility or are unable to use standard input devices, offers an alternative means of interaction that must only be managed by simple hand movement.

Collaborative Work: Within meeting room or classroom settings, could enable people to collaborate on joint digital whiteboards using natural motions, perhaps even integrating with video conferencing apps.

Creative Uses: Artists and designers might be able to use the intuitive drawing mode, possibly enhanced by AI-driven creative ideas or style transfer.

Limitations and Challenges

There were certain limitations encountered in our assessment:

Environmental Dependencies: It is dependent on lighting, camera resolution, and background complexity. Accuracy in gesture recognition may be affected by low light.

Hand Occlusion: When the hands overlap or move out of the camera's field of view, the system will lose tracking for a while, leading to discontinuities in drawing tasks.

AI Interpretation Limits: While Gemini AI possesses amazing powers of drawing interpretation, it also misinterprets complex drawings or ambiguous content on occasion, particularly when different components interact with one another on the canvas.

Physical Fatigue: Prolonged use of mid-air gestures can cause arm fatigue, more commonly referred to as "gorilla arm syndrome" [17]. This may restrict the system's usability over a long time without ergonomic adjustment.

7. FUTURE WORK

According to our results and determined limitations, we suggest some future research and development directions:

Expanded Gesture Vocabulary: Having the system recognize more gestures would enable more sophisticated interactions and operations. These might include two-hand gestures to perform operations like scaling or rotation.

Adaptive Gesture Recognition: Making the gesture patterns of each user adaptive with the help of machine learning techniques would improve the recognition rate and make the interaction personalized.

3D Interaction: The system extension to handle depth information can enable three-dimensional modeling and drawing applications.

Multimodal Interaction: Combining voice commands and gesture input would provide an integrated and natural interaction model, allowing users to explain complex operations in words and gesture to manage spatial controls.

Improved Feedback Mechanisms: Tactile or visual feedback can improve user awareness of system state and gesture detection, and possibly accuracy and error minimization.

Cross-Device Collaboration: Enabling several users to collaborate on one shared canvas simultaneously, quite possibly across geographies and devices, could enhance collaborative apps.

8. CONCLUSION

Based on our findings and discovered limitations, we propose several directions of future development and research:

Enhanced Gesture Vocabulary: Adding more gestures to the system so that it can support more types of gestures would enable more expressive interaction and functionality. Examples could include two-hand gestures to control scaling or rotating. Adaptive Gesture Recognition: Using machine learning techniques to learn from a user's gesture patterns would improve recognition accuracy and enable personalization of the interaction experience. 3D Interaction: Adding the ability to capture depth information would allow three-dimensional drawing and modeling software. Multimodal Interaction: Blending voice commands with gesture input can provide a more natural and richer interaction model, where users can specify complex operations with words and spatial information with gestures. Enhanced Feedback Mechanisms: Visual or haptic feedback may enhance user sensitivity to system status and gesture recognition, enhancing accuracy and diminishing errors. Cross-Device Collaboration: Allowing many users to engage with a shared canvas at the same time, possibly across disparate devices and geographies, could increase collaborative applications.

9. REFERENCES

- [1] P. K. Verma and R. A. Ambad, "Advancements in human-computer interaction: A comprehensive survey," IEEE Access, vol. 9, pp. 15089-15117, 2021.
- [2] A. Kumar, R. Singh, and S. K. Sharma, "Hand gesture recognition based human-computer interaction: A review," Int. J. Adv. Res. Comput. Sci. Software Eng., vol. 7, no. 5, pp. 146-151, 2017.
- [3] F. Zhang et al., "MediaPipe Hands: On-device real-time hand tracking," in Proc. Comput. Vision Pattern Recognit. (CVPR), 2020, pp. 1302-1311.
- [4] S. Brown et al., "Language models are few-shot learners," in Advances in Neural Information Processing Systems, 2020, vol. 33, pp. 1877-1901.
- [5] M. Saarinen, "Cryptanalysis of block ciphers based on SHA-1 and MD5," IACR Journal of Cryptography, 2021.
- [6] S. Kumar, "Improved cryptographic technique using SHA-1 and Merkle-Damgard construction," Think India Journal, 2020.
- [7] "Real-time hand gesture detection using ML," AI and Robotics Journal, 2022.
- [8] "AI-powered drawing assistance and feedback," Journal of AI Applications, 2020.
- [9] "Enhancing user interaction with AI and gesture," IEEE Transactions on Human-Computer Interaction, 2023.
- [10] "Multi-modal gesture recognition for interactive systems," in International Conference on AI & Virtual Reality, 2022.
- [11] "Gesture-based interaction for interactive apps," in ACM SIGCHI Conference, 2021.

-
- [12] I. S. MacKenzie and C. Ware, "Lag as a determinant of human performance in interactive systems," in Proc. INTERACT Conf. Human Factors Comput. Syst., 1993, pp. 488-493.
 - [13] J. Brooke, "SUS: A 'quick and dirty' usability scale," in Usability Evaluation in Industry, London, UK: Taylor & Francis, 1996, pp. 189-194.
 - [14] D. Hincapié-Ramos et al., "Consumed endurance: A metric to quantify arm fatigue of mid-air interactions," in Proc. SIGCHI Conf. Human Factors Comput. Syst., 2014, pp. 1063-1072.