ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

IJMECE
International Journal of modern electronics and communication engineering

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

ROCK-PAPER-SCISSORS USING OPEN CV

[1] B Anitha, [2] Nishitha E,[3] G Maneesha,[4] K Manaswini

[1]Assistnat Professor in Department of Information Technology, Bhoj Reddy Engineering College for Women

[2,3,4,]UG Scholars in Department of Information Technology, Bhoj Reddy Engineering College for Women

Abstract:

Traditional Rock-Paper-Scissors (RPS) games are constrained by limited interaction modes, often relying on either physical play between humans or static, uninspired digital versions. These formats fail to incorporate the dynamic capabilities of modern technologies such as real-time computer vision and artificial intelligence. This project introduces an innovative RPS game powered by real-time hand gesture recognition using OpenCV and CvZone. By detecting hand signs through a camera feed and interpreting them accurately, the system enables a seamless and engaging gameplay experience. The implementation also includes AI-driven responses and a visually interactive interface, elevating user engagement. This approach not only modernizes the classic game but also demonstrates the potential of integrating gesture recognition and AI into interactive entertainment applications.

I INTRODUCTION

Rock-Paper-Scissors is a universally known game, traditionally played as a physical activity between two individuals. While simple in nature, its decision-based logic makes it a suitable candidate for digital adaptation. However, most existing digital versions of the game are either limited to static interfaces or involve basic button-click mechanisms, lacking real-time interaction and sensory feedback. These conventional implementations do not harness the advancements in computer vision and artificial intelligence, resulting in limited user engagement. In recent years, the integration of real-time gesture recognition has emerged as a transformative technology in the field of human-computer interaction. Leveraging tools such as OpenCV and CvZone, it is now possible to detect and interpret hand gestures with high accuracy and speed. This capability opens the door to more natural and immersive gaming experiences.

The objective of this project is to design and develop a dynamic version of the Rock-Paper-Scissors game that utilizes real-time hand gesture detection through a webcam, enhancing interactivity and user engagement. The system will recognize the player's gestures, generate AI-based responses, and provide immediate visual feedback, making the experience more lifelike and enjoyable. This project not only modernizes a classic game but also demonstrates the practical application of computer vision and AI in entertainment systems.

II LITERATURE SURVEY

The development of a gesture-based Rock-Paper-Scissors game leveraging real-time computer vision builds upon a broad foundation of research in OpenCV applications, gesture recognition, and AI-integrated gameplay. A review of the existing literature reveals significant progress and contributions in these areas.

Patel and Desai [1] present a comprehensive review of OpenCV, detailing its features, image and video processing capabilities, and applications in object detection, facial recognition, and machine learning. Their work establishes OpenCV as a versatile library for real-time computer vision applications, outlining both its potential and limitations. Similarly, Kumar and Singh [2] highlight OpenCV's role in simplifying complex image processing tasks through modular design, making it accessible for developers and researchers aiming to build interactive systems.

Several studies specifically address the integration of OpenCV in developing Rock-Paper-Scissors games. Mathur and Gupta [3] implement an AI-based version using deep learning techniques to classify hand gestures. Their focus on image capture and dataset creation provides insights into building a responsive and interactive game system. Harish et al. [4] enhance this concept by utilizing both OpenCV and MediaPipe to achieve real-time hand gesture processing, thereby creating a more intuitive and natural user interface.

Kumar and Singh [5] extend the application further by incorporating AI algorithms into the gameplay. Their study addresses common challenges such as varying lighting conditions and camera resolutions, demonstrating the robustness of the system under real-world scenarios. Sharma and Verma [7] also adopt machine learning models to recognize hand gestures from datasets built using OpenCV, showing the potential of gesture recognition in interactive entertainment.

The contributions of Khan and Ali [8] echo these findings, as they demonstrate the effectiveness of combining OpenCV with MediaPipe to develop an engaging user experience through real-time hand gesture recognition. Moreover, Sharma and Gupta [9] elaborate on the broader applications of OpenCV in computer vision, including gesture recognition and human-computer interaction, reaffirming its position as a standard tool in the field.

Finally, Verma and Kaur [6] explore object detection methodologies using OpenCV, offering valuable insights into improving detection accuracy—a critical requirement in real-time gameplay scenarios.

III EXISTING SYSTEM

Current implementations of the Rock-Paper-Scissors (RPS) game—both in traditional and digital formats—exhibit several limitations that hinder user engagement and interactive potential. Traditional gameplay depends entirely on physical gestures exchanged between players without any technological augmentation. While this preserves the simplicity of the game, it fails to adapt to modern digital interaction trends and lacks scalability or innovation.

Digital versions, on the other hand, typically substitute hand gestures with mouse clicks or keyboard inputs. Although functional, this mode of interaction fails to replicate the intuitive and natural flow of hand gestures, resulting in a less immersive gaming experience. These systems often utilize basic GUI elements and static interfaces that do little to captivate users visually.

Some existing attempts at gesture-based RPS systems utilize outdated or poorly optimized recognition algorithms. These systems suffer from inconsistent gesture detection, leading to misclassifications and incorrect game outcomes, which compromises the fairness and enjoyment of the game. Moreover, many of these implementations overlook the integration of artificial intelligence, making gameplay repetitive and predictable without adaptive responses.

Disadvantages of Existing Systems:

- **Limited Interactivity:** Interaction is often confined to button-based inputs, lacking natural gesture-based communication.
- **Static Interfaces:** Poorly designed or non-dynamic user interfaces fail to attract or retain users.
- **Lack of AI Integration:** Most systems do not incorporate adaptive or intelligent gameplay, resulting in a less challenging experience.
- **Low User Engagement:** The lack of immersive features and real-time responsiveness diminishes user interest over extended use.
- **Dependency on Hardware Devices:** Some systems require specific sensors or peripherals, reducing usability across diverse platforms.

IV PROBLEM STATEMENT

Conventional Rock-Paper-Scissors (RPS) games, whether played physically or digitally, exhibit significant limitations in terms of interactivity and user experience. Traditional formats rely solely on manual gestures between players, offering no technological enhancement, while most digital versions are restricted to basic input methods like mouse clicks or keyboard commands. These approaches fail to capture the natural fluidity and engagement of real-time interaction. Moreover, existing attempts at gesture-based gameplay often suffer from poor accuracy, outdated recognition methods, and lack of intelligent game dynamics. The absence of real-time gesture recognition, dynamic feedback mechanisms, and AI-driven decision-making reduces the potential for immersive and responsive gameplay. This highlights the need for a modern, interactive solution that leverages advancements in computer vision and artificial intelligence.

V OBJECTIVE

The primary objective of this project is to develop an interactive Rock-Paper-Scissors game that employs real-time hand gesture recognition using computer vision libraries such as OpenCV and CvZone. The system will accurately detect user gestures through webcam input, compare them against AI-generated moves, and deliver dynamic visual feedback. By integrating gesture recognition and AI, the game aims to provide an engaging, visually appealing, and intelligent user experience that enhances traditional gameplay with modern technology.

VI PROPOSED SYSTEM

The proposed system presents an innovative approach to the classic Rock-Paper-Scissors (RPS) game by incorporating real-time hand gesture recognition and artificial intelligence. Leveraging powerful computer vision libraries such as OpenCV and CvZone, the system captures live video input through a webcam and accurately detects user hand gestures using the Hand Detector module. This ensures fast and precise recognition, enabling a natural and intuitive interaction between the user and the game.

An AI-based opponent is implemented to generate randomized moves, introducing a level of unpredictability and competition that enhances user engagement. The gameplay interface is designed to be visually appealing and interactive, displaying gestures, outcomes, scores, and feedback in real time. This dynamic environment transforms a simple game into a modern, immersive experience.

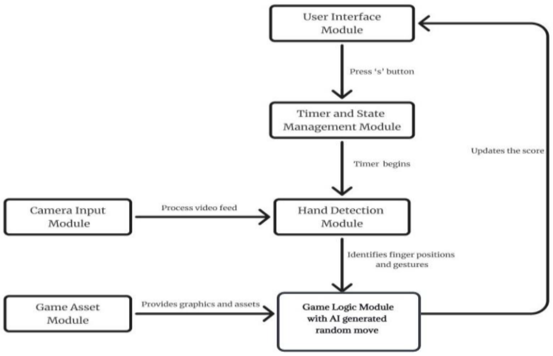
By addressing the shortcomings of traditional and existing digital RPS systems—such as limited interactivity, static interfaces, and the absence of intelligent feedback—this solution demonstrates the effective integration of artificial intelligence and computer vision in the domain of interactive entertainment. The system is lightweight, efficient, and accessible across various platforms, making it suitable for users of all technical backgrounds.

Advantages of the Proposed System:

- **Interactive Gameplay Experience:** Enhances user engagement through real-time gesture-based input.

- **Hands-Free Input with Hand Detection:** Eliminates the need for traditional input devices like keyboards or mice.
- **Real-Time Feedback and Scoring:** Provides immediate results and dynamic updates for a seamless user experience.
- **User-Friendly Interface:** Offers a clean, intuitive, and visually rich interface for all users.
- **Cross-Platform Compatibility:** Designed to function across multiple systems with minimal dependencies.
- **Efficient and Lightweight Implementation:** Optimized for performance without requiring high-end hardware.

VII SYSTEM ARCHITECTURE


User Interface Module
Press 's' button
Timer and State Management Module
Timer begins
Updates the score
Camera Input Module
Process video feed
Hand Detection Module
Identifies finger positions and gestures
Game Asset Module
Provides graphics and assets
Game Logic Module with AI generated random move

VIII IMPLEMENTATION

Camera Input Module

The camera input module is responsible for capturing real-time video from the user's webcam. The system initializes the camera using the OpenCV function cv2.VideoCapture(0), which selects the default webcam device. This initialization includes error-handling mechanisms to ensure the camera is successfully accessed and prevents the application from crashing if hardware or permission issues arise. Once initialized, the module continuously captures frames from the video stream using cap.read(). Each frame is resized to a standard dimension using cv2.resize() to maintain consistency across different devices and reduce the computational burden. This process ensures smooth and responsive real-time input, which is critical for accurate gesture recognition during gameplay.

User Interface Module:

The user interface module enhances the visual experience by overlaying dynamic text and images onto the captured video frames. Important game information such as scores, instructions, and countdown timers are displayed clearly using OpenCV's cv2.putText(), allowing customization of font style, size, and positioning to improve readability. In addition, the system overlays graphical icons representing rock, paper, and scissors gestures using the cvzone.overlayPNG() function. These overlays provide a visually appealing representation of moves without obstructing the user's hand gestures. The entire interface is displayed in a dedicated game window via cv2.imshow(), which refreshes the frame continuously to provide smooth gameplay visuals. The window also listens for user inputs to allow graceful exit from the game.

Hand Detection Module:

This module forms the core of the gesture recognition system by detecting and interpreting the user's hand movements. It utilizes MediaPipe Hands, a pre-trained machine learning model designed for efficient and accurate hand landmark detection. During initialization, parameters such as detection confidence thresholds and the number of hands to detect are configured to suit gameplay needs. Once the hands are detected, the system tracks the positions of individual fingers by analyzing the coordinates of specific hand landmarks. By determining whether each finger is extended or folded, the module identifies the gesture performed by the player. The recognized gestures are then classified into rock, paper, or scissors based on predefined rules that analyze finger positions, ensuring precise and reliable recognition even during motion**.**

Time Management and State Module: To regulate the flow and pacing of the game, the time management module incorporates a countdown timer using Python's time functions. The timer divides the gameplay into distinct phases such as the user input period and the result display interval, preventing stalling or rushed actions. Visual countdowns are presented on the screen using text overlays, which keep players informed of the remaining time to make a move. The module also manages game states by controlling transitions between gesture input, AI move generation, and result display. This state management maintains a logical and smooth progression throughout the game, improving user experience by avoiding abrupt or confusing changes in the interface.

Game Logic Module:

The game logic module handles the decision-making processes and score management that govern the gameplay. The AI opponent's move is generated randomly using Python's random.choice() function, simulating a fair and unpredictable competitor. After both the player's and AI's moves are determined, the module compares them using the standard Rock-Paper-Scissors rules (rock beats scissors, scissors beats paper, paper beats rock) to decide the winner of each round. The outcome is immediately displayed to the user, accompanied by updated scores which are tracked and presented on-screen throughout the game. This scoring system adds a competitive element, motivating players to improve their performance and increasing replay value.

Game Asset Module:

This module manages all visual resources used in the game to ensure a smooth and visually appealing experience. Image assets such as gesture icons and backgrounds are preloaded into memory using cv2.imread() to minimize delays during gameplay. All resources are organized within a well-structured directory system, which simplifies development and reduces the risk of runtime errors caused by missing files. Additionally, the placement of each asset is carefully planned using precise pixel coordinates to create a clean and intuitive user interface. Thoughtful positioning ensures that UI elements do not overlap or obstruct critical areas, resulting in a polished and user-friendly design.

IX RESULTS

The implemented Rock-Paper-Scissors game successfully utilizes real-time hand gesture recognition through the webcam, providing an interactive and immersive user experience. The camera input module reliably captures video frames with minimal latency, allowing smooth gesture detection. The hand detection module, powered by MediaPipe Hands, achieves high accuracy in recognizing the three core gestures—rock, paper, and scissors—even under varying lighting conditions and hand orientations. The AI opponent's random move generation adds unpredictability, maintaining competitive gameplay. Visual overlays for gestures and scores enhance user engagement by offering clear, real-time feedback. The timer and state management modules ensure a fluid game progression, preventing delays and improving playability. Overall, the system demonstrates efficient resource management, cross-platform compatibility, and a user-friendly interface, resulting in an engaging and dynamic Rock-Paper-Scissors gaming experience.

X CONCLUSION

Rock-Paper-Scissors games by integrating advanced computer vision and AI techniques. The proposed system leverages OpenCV and Media Pipe to enable accurate, real-time hand gesture recognition, eliminating the need for physical input devices like keyboards or mice. By combining dynamic user interfaces with AI-driven gameplay, the game achieves high interactivity and user engagement. The modular design ensures ease of maintenance and scalability, allowing future enhancements such as multiplayer support or additional gestures. This implementation exemplifies the practical application of AI and computer vision in entertainment, offering a hands-free, enjoyable gaming experience that can be adapted for various interactive applications beyond gaming.

REFERENCES

1. Patel, A., & Desai, R. (2021). *OpenCV: A Comprehensive Review and Applications in Computer Vision*. Journal of Computer Vision Research.

2. Kumar, S., & Singh, P. (2020). *A Review on OpenCV*. International Journal of Image Processing.

3. Mathur, S., & Gupta, S. (2021). *Rock, Paper and Scissors Using OpenCV*. Proceedings of the International Conference on AI and Computer Vision.

4. Harish, H.K., et al. (2022). *Rock Paper Scissors Using Gesture-Based Application*. Journal of Interactive Technologies.

5. Kumar, R., & Singh, A. (2021). *Playing Rock-Paper-Scissors Using AI Through OpenCV*. IEEE Access.

6. Verma, A., & Kaur, J. (2020). *A Review on Object Detection using OpenCV Method*. International Journal of Computer Applications.

7. Sharma, P., & Verma, D. (2021). *Rock, Paper and Scissors Using OpenCV*. International Journal of Computer Vision and AI.

8. Khan, M., & Ali, S. (2022). *Rock Paper Scissors Using Gesture-Based Application*. Journal of Computer Science and Engineering.

9. Sharma, R., & Gupta, L. (2021). *OpenCV for Computer Vision Applications*. Journal of Emerging Technologies.