iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.70602

AIR-CANVAS: A Computer Vision Based Gesture Drawing System

Kiran Ande[1], Anushka Korde[2], Deepak Dhale[3], Kartik Gujarkar[4], Hrutik Hiwase[5]
[1]Assistant Professor, [2, 3, 4, 5, 6]UG Students, Department of Information Technology, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India

Abstract: The Air Canvas Project is an innovative computer vision application developed using OpenCV and Python, which enables users to draw in the air using simple hand gestures tracked via a webcam. By utilizing a coloured marker—typically placed at the tip of a finger—the system captures the motion of the marker in real time and renders corresponding strokes on a virtual canvas. The project leverages key techniques in computer vision, including colour detection in HSV colour space, contour tracking, and morphological operations like erosion and dilation to enhance accuracy and reduce noise.

The system continuously tracks the position of the coloured marker, stores the coordinates, and draws the path traced by the user across successive video frames. This approach eliminates the need for traditional input devices, offering a touchless, interactive experience. Designed with accessibility and simplicity in mind, the Air Canvas can serve as both an educational tool for learning computer vision and a creative platform for digital expression. The project demonstrates the potential of combining real-time image processing and gesture recognition to create intuitive, user-friendly applications.

I. INTRODUCTION

The Air Canvas Project is a creative and interactive computer vision application that allows users to draw on a digital canvas by simply waving a finger in the air. This system utilizes a coloured marker, typically placed at the tip of the user's finger, and a web camera to track motion and convert it into real-time drawings. The concept removes the need for traditional input devices like a mouse, pen, or touch screen, offering a touchless, gesture-based drawing experience.

Developed using OpenCV and Python, the project leverages powerful computer vision techniques such as colour detection, contour analysis, and morphological image processing. The camera captures each frame, which is then converted into the HSV colour space—a format better suited for colour-based object tracking. Using these techniques, the system detects the marker, refines the output using erosion and dilation, and stores the position of the marker over time. These positions are then used to draw continuous lines on both the video feed and a separate canvas, effectively visualizing the user's motion as virtual ink strokes.

This project showcases how real-time computer vision can be applied to intuitive human-computer interaction, offering both an educational platform for learning OpenCV and a fun digital tool for creative expression.

II. LITERATURE SURVEY

Advancements in computer vision, real-time image processing, and user-interface design have led to the development of innovative applications that merge physical gestures with digital outputs. The Air Canvas Project falls under this evolving area, combining gesture recognition and object tracking to create an engaging and interactive system.

A. Key Concepts and Supporting Research:

1) Colour-Based Object Tracking Object tracking based on colour segmentation has proven effective due to its simplicity and efficiency, especially in real-time applications. By converting the input image to HSV colour space, the system enhances accuracy in colour detection under varying lighting conditions.
2) Morphological Operations in Image Processing Erosion and dilation are fundamental techniques used to reduce noise and refine detection masks, ensuring that only meaningful visual data (i.e., the coloured marker) is captured and tracked accurately.
3) Contour Detection and Real-Time Interaction Contour analysis allows the system to identify the largest object matching the colour mask. The center of this contour serves as the key tracking point, which is continuously monitored and stored to represent the user's drawing path.

B. Existing Applications and Studies:

1) Gesture-Controlled Interfaces: Research in human-computer interaction (HCI) demonstrates the potential of gesture recognition in replacing traditional input methods. Similar techniques are used in AR/VR systems, motion-based gaming, and smart device control.
2) Virtual Whiteboard Tools: Similar projects have been developed to turn gestures into writing or drawing tools, such as "virtual whiteboards" or "smart classroom systems," which often use OpenCV and similar tracking mechanisms.
3) Educational Value: This project type is frequently adopted in academic settings for introducing students to computer vision fundamentals, including image filtering, object localization, and interactive GUI development.

III. PROPOSED SYSTEM

The Air Canvas Project is a computer vision-based drawing system developed using OpenCV and Python. This system allows users to draw virtually in the air using a coloured marker, such as an object placed at the tip of a finger. By capturing the movement of the coloured marker through a camera, the system tracks the motion in real time and renders the path on a digital canvas.

The system primarily uses colour detection and tracking to identify the marker. It works by capturing video frames from the webcam and converting them into HSV colour space, which simplifies the process of identifying specific colours under different lighting conditions. Once the coloured object is detected, a mask is created, which is further refined using morphological operations—specifically erosion and dilation. These operations help eliminate noise (small unwanted areas) and restore the shape of the main object.

The processed image is then analysed to detect contours, from which the center of the largest contour (representing the marker) is found. This center point is stored in an array that keeps track of all points across frames. These points are then used to draw on both the live video feed and a dedicated canvas, creating the effect of drawing in the air.

A. System Algorithm

1) Capture frames from the webcam.
2) Convert frames to HSV colour space for effective colour-based tracking.
3) Prepare the canvas and add virtual buttons for different ink colours.
4) Adjust HSV values using trackbars to detect the marker colour and create a binary mask.
5) Apply morphological operations:
   - Erosion to remove small noise in the mask.
   - Dilation to restore the eroded parts of the main object.
6) Detect contours in the mask.
7) Find the center of the largest contour.
8) Store the center points in an array to track movement over time.
9) Draw on the canvas and video feed using the stored points.

B. System Requirements

1) Python 3
2) OpenCV
3) NumPy

IV. EXISTING SYSTEM

In traditional drawing systems, users must physically interact with a drawing tool like a pen, stylus, or touchscreen to create an image or sketch. These methods require direct contact with a surface, such as paper or a screen, limiting the interaction to only those with a tangible medium. Additionally, such systems often depend on hardware like drawing tablets or touchscreens, which are relatively expensive and require specific devices for use.

The existing systems for drawing, including tablet-based drawing and stylus inputs, also fail to provide a gesture-based or air drawing experience, where users can simply move their hands or fingers in space and produce a drawing.

The Air Canvas Project introduces an innovative way to draw using computer vision and gesture recognition. By utilizing a coloured marker attached to the tip of a finger, the system can track the motion of the marker through a regular webcam.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

This enables users to draw in the air without touching any surface, making the drawing process entirely gesture-based and highly interactive.

Unlike traditional systems that are constrained by physical tools and screens, the Air Canvas Project employs advanced colour detection and tracking techniques to capture the movement of a coloured object, using the power of OpenCV and Python. The system tracks the marker in real-time, drawing corresponding strokes on a canvas by simply detecting the movement of the user's finger in the air.

Key Features of the Existing System:

1) Camera-Based Tracking: Traditional drawing systems rely on direct contact with a screen or surface, while the Air Canvas Project uses a webcam to track the motion of a coloured marker (such as a coloured object at the tip of the user's finger).
2) Real-Time Drawing: The system processes the motion of the coloured marker in real time and translates it into digital strokes on a canvas.
3) OpenCV Integration: OpenCV is used to handle colour detection, contour tracking, and image processing to create the drawing.
4) Morphological Operations: The system uses operations like Erosion and Dilation to refine the mask and ensure smooth and accurate detection of the coloured marker.
5) Canvas Output: The system continuously updates a digital canvas with the drawn strokes, providing an intuitive, touchless drawing experience.

V. MODULES AND FUNCTIONALITIES

The Air Canvas Project can be divided into distinct modules, each handling a specific functionality essential for real-time gesture-based drawing. Below is a breakdown of the core modules and their corresponding functionalities for building the Air Canvas system using OpenCV and Python.

A. *User Interaction and Tracking Module*

◈ Functionalities:

1) User Input via Finger Gesture: The system uses camera input to capture the user’s finger motion, where the coloured marker on the tip of the finger is the primary object being tracked.
2) Real-Time Tracking: Using OpenCV, the system tracks the motion of the coloured marker, which acts as the drawing tool for the user. The camera captures the video feed, and the system processes each frame for the position of the coloured marker.
3) HSV Colour Detection: Frames captured by the camera are converted to HSV colour space, as it is easier to isolate a specific colour and track its position in the frame. The system filters out everything except the selected colour (e.g., a bright coloured object).

B. *Masking and Contour Detection Module*

◈ Functionalities:

1) Mask Generation: Once the coloured marker is detected, the system generates a binary mask isolating the area of interest (the marker). This helps eliminate irrelevant data from the image.
2) Morphological Operations: The mask goes through Erosion and Dilation processes to remove small impurities (such as noise) and restore the shape of the detected object. Erosion shrinks the mask, while dilation expands it, ensuring accurate detection of the marker.
3) Contour Detection: The system detects contours within the mask. These contours correspond to the boundaries of the coloured marker, and from these, the center of the marker is calculated. This center point is stored for future drawing on the canvas.

C. *Canvas and Drawing Module*

◈ Functionalities:

1) Digital Canvas Creation: A canvas is created as the backdrop where the drawing will appear. The user can use their finger as a pen to draw on this digital canvas, similar to a traditional drawing surface.
2) Drawing the Points: The coordinates of the detected center of the coloured marker are collected over time and used to draw points on the canvas in real-time. These points are drawn with a certain brush size or pen style to create fluid, continuous lines.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

3) Dynamic Stroke Rendering: As the user moves their finger, the system draws corresponding strokes on the canvas, creating shapes, lines, and figures. This allows for freehand drawing without direct touch.

D. Interaction Controls Module

- Functionalities:

1) Adjusting Parameters: The system provides a simple interface with trackbars that let the user adjust parameters like colour range, pen size, or stroke thickness in real time.
2) Button Controls for Ink Options: The canvas can include buttons for ink selection, allowing the user to switch between different drawing tools, such as pen, eraser, and more.
3) Clear Canvas Option: Users can easily clear the canvas to start over or reset the drawing area.

E. System Optimization and Performance Module

- Functionalities:

1) Real-Time Performance: The system is optimized for low latency and high performance, enabling the camera feed and the drawing process to occur with minimal delay.
2) Motion Detection Filtering: The system includes filtering algorithms that ignore small, insignificant movements of the hand, ensuring that only intentional gestures result in drawing on the canvas.
3) Customizable Sensitivity: Users can adjust the sensitivity of the system to detect larger or more delicate movements, depending on their preference for drawing accuracy.

F. Requirements and Dependencies Module

- Functionalities:

1) Python Libraries: The project relies on Python3, NumPy, and OpenCV. These libraries provide all the essential tools for image processing, motion tracking, and drawing on the canvas.
2) Hardware Requirements: The system requires a standard webcam or any camera module that supports real-time video capture.
3) Platform Requirements: The application runs on any system with the necessary libraries installed, including Windows, macOS, and Linux.

G. Safety & System Integrity Module

- Functionalities:

1) Error Handling: The system includes error handling to ensure it continues to function smoothly in case of camera issues or detection failures.
2) System Reset: Users can reset the drawing area or adjust detection parameters without restarting the entire system, ensuring a seamless experience.

VI. RESULTS

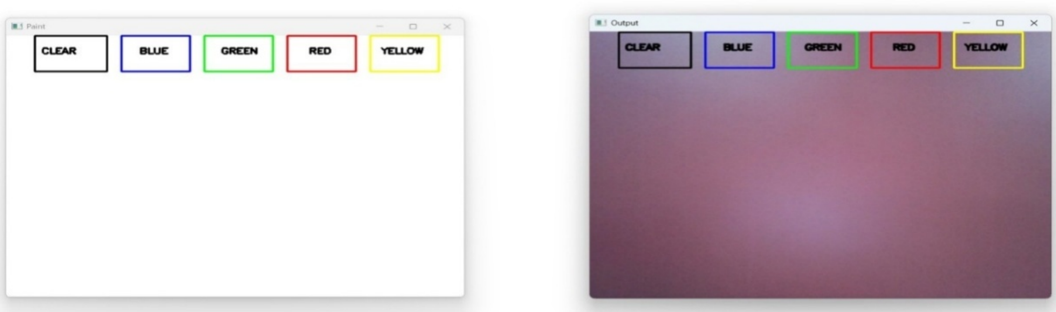

Fig.no 1. The initial Air-Canvas interface waiting for user input: The system is ready for interaction, displaying colour selection buttons (CLEAR, BLUE, GREEN, RED, YELLOW) at the top of both the canvas and camera feed windows. With no hand detected, the camera feed remains idle, demonstrating the system's real-time responsiveness to hand presence and movement.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

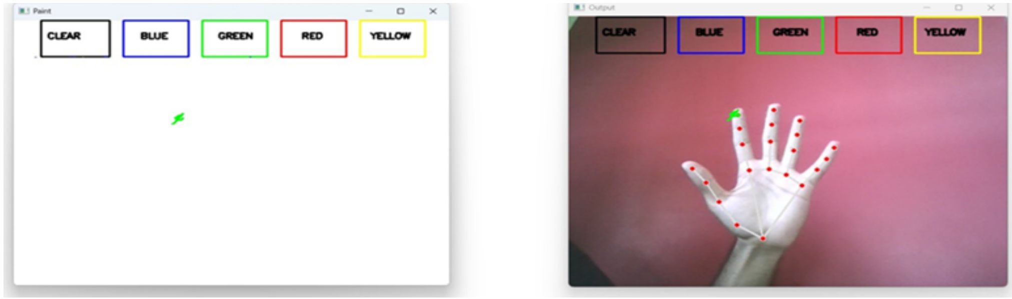

Fig.no 2. An interactive Air-Canvas system in action: Using real-time hand tracking and a green fingertip marker, the user draws in the air without touching the screen. The computer vision model detects the fingertip position and translates it onto a virtual canvas, enabling colour selection and drawing through gesture-based control using OpenCV in Python.

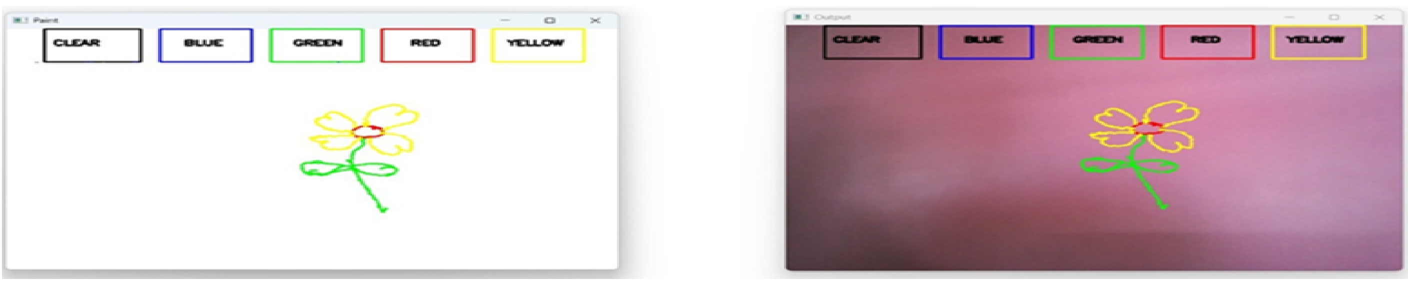

Fig.no 3. Gesture-based colour selection in the Air-Canvas project: The system recognizes a hand gesture using real-time landmark tracking to switch drawing colours. With the fingertip marker aligned over the 'GREEN' box, the user successfully selects green as the active drawing colour. This intuitive, touchless interaction is enabled by OpenCV and Python's powerful computer vision capabilities.

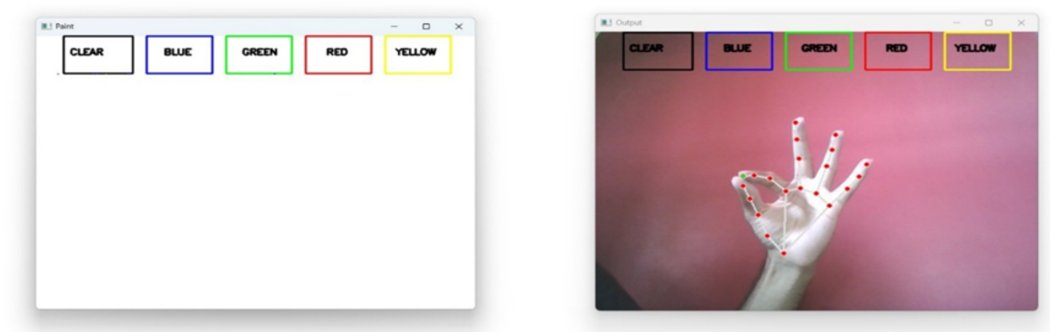

Fig.no 4. Real-time virtual drawing using Air-Canvas: The user draws a flower on the digital canvas using hand gestures tracked by the webcam. The interface accurately recognizes finger positions to render different parts of the flower in selected colours (green for the stem and leaves, yellow for petals, red for the center), showing the seamless interaction between gesture control and drawing output.

VII. CONCLUSION

The Private Car Rental Web Portal successfully showcases how digital transformation can revolutionize the traditional car rental industry by offering a secure, peer-to-peer platform. This solution provides significant benefits to both vehicle owners and renters, such as streamlined booking processes, a robust verification system, and flexible rental options. The system's user-friendly interface and automation reduce administrative tasks and ensure transparent transactions for all users.

A standout feature of the platform is its contribution to sustainable transportation. By optimizing vehicle utilization, it helps reduce the environmental impact associated with private car ownership, promoting the sharing economy.

The project has successfully met its key objectives, proving the technical feasibility of web-based rental platforms. Furthermore, it highlights the potential to reshape urban mobility. Future improvements, such as AI-powered dynamic pricing, real-time GPS tracking, and enhanced mobile functionalities, promise to further elevate the user experience. The continuous refinement of the system, driven by user feedback and technological integration, will ensure its relevance and competitiveness in the ever-evolving shared economy landscape. This platform sets a benchmark for future developments in the car rental industry.

VIII. FUTURE ENHANCEMENTS

To further improve the system and enhance the user experience, several features can be integrated in future updates:

A. GPS Tracking & Real-Time Monitoring

1) Real-Time Location Tracking: Integrating GPS tracking would enable users to monitor the car's location during the rental period, providing peace of mind for both owners and renters.
2) Geo-Fencing: This feature would restrict the car's movement within designated areas, adding an extra layer of security and ensuring that the car remains within approved zones.
3) Route History Tracking: Vehicle owners could track the route history of their rented vehicles, offering insights into their usage and ensuring that the cars are used responsibly.
4) Nearby Car Finder: Renters could use GPS functionality to find the nearest available rental cars, streamlining the booking process and reducing wait times.

B. AI-Based Pricing & Recommendations

1) Dynamic Pricing Algorithm: By using AI-based dynamic pricing, the system could adjust the car rental prices based on factors such as demand, season, and location. This would optimize pricing for both owners and renters and increase system efficiency.
2) AI-powered Recommendations: The system could suggest the most suitable cars based on users' preferences and history, improving the overall booking experience. For example, if a user frequently rents SUVs, the system would prioritize similar options in future searches.
3) Personalized Discounts: Based on rental history and frequency, the platform could offer personalized discounts to frequent renters, encouraging customer loyalty and enhancing user retention.

These enhancements would not only improve the platform's functionality but also create a more intuitive and efficient user experience, ensuring the system remains a top choice for car rentals in the competitive market.

REFERENCES

[1] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.A foundational paper on the OpenCV library, used for real-time computer vision.
[2] Rosebrock, A. (2021). Real-Time Object Tracking with OpenCV and Python. PyImageSearch.A detailed tutorial on colour tracking and object detection using OpenCV.
[3] Zhang, Y., & Liu, H. (2020). Vision-Based Gesture Recognition Using Machine Learning Techniques: A Review. Sensors, 20(14), 3910.A review of gesture recognition methods with applications in HCI.
[4] Sharma, A., & Bhardwaj, A. (2021). Virtual Drawing using Finger Detection and Tracking. International Journal of Computer Applications, 183(43).Implementation of a similar finger-drawing system using computer vision.