IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ※ Peer-reviewed & Refereed journal ※ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141218

"Futuristic Digital Art: AI-Driven Painting with Gesture & Automated Shape Precision"

Sowbhagya M P[1], Adoni Anirudh[2], Ashish Reddy V P[3], Balaji R[4], K M Thejdeep Krishna[5]

Associate Professor, Dept. of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru[1]

Student, Dept. of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru[2]

Student, Dept. of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru[3]

Student, Dept. of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru[4]

Student, Dept. of Computer Science and Engineering, K.S. Institute of Technology, Bengaluru[5]

Abstract: The development of human-computer interaction (HCI) is constantly pushing the limits of digital art creation. Despite this, the art of creating digital art continues to rely heavily on conventional input devices such as the stylus and the mouse, which may be a deterrent to many. This paper presents "Futuristic Digital Art," a revolutionary AI-powered painting software that turns an ordinary webcam into a smart and emotive creative artistic device. The system uses real-time vision-based hand tracking to provide natural gesture controls for brush and color selection, drawing, and other operations on the canvas. The core of its innovation is a sophisticated auto-correction engine capable of detecting and perfecting hand-drawn geometric shapes and letters with high accuracy. By utilizing the MediaPipe framework for reliable hand landmark detection and an algorithmic solution for gesture understanding and shape identification, the system provides an intuitive, hardware-independent solution. This work shows an important advancement towards democratizing digital art and interaction by combining the expressiveness of natural human motion with the accuracy of smart computational guidance.

Keywords: AI Painting, Gesture Recognition, Human-Computer Interaction, Digital Art, MediaPipe, Shape Correction, Computer Vision.

I. INTRODUCTION

The move from physical canvas to virtual screen has made art more democratized, but the ways of interacting haven't kept up. Most art software uses indirect input devices that fail to provide the instantaneity and organicity of their physical counterparts. This disconnects the creator's idea from the digital result. Gesture-based interfaces offer an interesting alternative with the possibility of a more natural and engaging creative experience.

"Futuristic Digital Art: AI-Powered Painting with Gesture & Automated Shape Accuracy" is an innovative project aimed at closing this gap. It leverages the capabilities of contemporary computer vision and artificial intelligence to provide a rich digital painting experience without needing any advanced hardware other than a basic webcam. The system employs sophisticated hand tracking technology to decode a user's gestures, converting them to commands and brush strokes on a digital painting surface.

The key contribution of the project is a two-pronged intelligent assistance approach. Firstly, it uses a novel two-hand control paradigm in which one hand performs commands (e.g., enabling draw mode, choosing a tool) and the other hand carries out the action (e.g., painting, choosing a color). Secondly, it employs a highly effective real-time correction system. This engine processes the freehand strokes of the user and, upon startup, automatically regularizes them into flawless geometric figures (circles, squares, triangles) or normalizes them into neat, readable letters. This marriage of intuitive control and smart precision provides a seamless workflow, allowing both inexperienced users and seasoned artists to bring their creative vision to life with unprecedented ease and accuracy.

II. EXISTING SYSTEM

The digital art and gesture control landscape is heterogeneous, with current solutions differing significantly in terms of accessibility, price, and capability.

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141218

•**Classic Digital Art Software:** Tools like Adobe Photoshop, Corel Painter, and Krita are industry stalwarts, providing incredible power and flexibility. But they interact mostly through a mouse or a graphics tablet. While there is more precision with tablets, both share a steep learning curve and are not as direct as free-hand motion.

•**Advanced Gesture Control Hardware:** Devices like Leap Motion and advanced VR interfaces like Tilt Brush enable extremely interactive and immersive 3D creative spaces. The technologies are good at interpreting subtle hand gestures as digital movements but are held back by the need for high-cost, specialty hardware, thus limiting their universal usage.

•**Early Vision-Based Interfaces:** Earlier academic and commercial efforts in webcam-based gesture control tended to use more rudimentary computer vision methods, such as background subtraction or color markers. These techniques also tended to suffer from low accuracy, lighting sensitivity, and a small vocabulary of gestures that they could recognize, which made them impractical for precision tasks such as drawing.

•**AI-Driven Generative Art:** Recent advances in AI have given rise to strong generative models (e.g., DALL-E, Midjourney) capable of generating intricate art from text inputs. Transformative as they are, these applications center around the product, removed from the process of creation. Our system is essentially dissimilar; it is not a content generation platform but an intelligent assistant that helps and augments the user's own creative act, retaining the artist in control of the process.

Our system overcomes all these limitations and makes a special niche in our proposed system. It provides an accurate, stable, and feature-laden gesture-based painting experience that is software-based, accessible by anyone who has a computer and a webcam.

III. METHODOLOGY

The system architecture of our system is that of a modular pipeline that takes visual input from the user and converts it into digital art in real-time. The central components are Hand Landmark Detection, Gesture Interpretation, a Drawing & Correction Engine, and UI Rendering.

3.1. System Workflow

The flow of operations, as depicted in the figure below, starts with recording video input from the user. The frame is processed to detect hand landmarks, which are then examined to infer what the user wants to do—draw, pick a tool, or give a command. Depending on this interpretation, the system updates the canvas and gives feedback to the user, and the process repeats in a seamless interactive loop.

• **Step 1**: Frame Capture: High-framerate video is captured from a regular webcam.

•**Step 2**: Hand Landmark Detection: The MediaPipe Hands model processes each frame to detect 21 main landmarks for every recognized hand.

•**Step 3:** Gesture Analysis: A spatial algorithmic interpreter examines the spatial relationships and configuration of these landmarks to identify gestures. This module identifies the command hand and the brush hand, and identifies the current system state (e.g., 'Draw', 'Select', 'Forced Shape').

• **Step 4:** Action Execution:

O During 'Draw' mode, the brush hand's index finger trajectory is captured as a stroke.

OThe position of the brush hand's index finger over the UI items in 'Select' mode defines the selected color or brush size.

O If a shape-forcing gesture is ongoing, the system pre-empts the correction engine.

•**Step 5:** Stroke Processing & Correction: Once a stroke is completed, the system passes the path data to the Drawing & Correction Engine. If auto-correction is enabled, the stroke is processed and replaced with an improved shape or letter. Else, the raw stroke is stored.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.141218

• **Step 6:** Canvas Rendering: The last brush stroke is drawn onto the permanent canvas, and the new frame, including UI overlays, is shown to the user.

3.2. Hand Tracking and Gesture Recognition

The basis of our system is its capacity to correctly monitor and comprehend hand movements. We utilize Google's MediaPipe framework, widely used for being very fast and accurate at real-time hand landmark detection. This enables our system to run at low latency, which is very important for an efficient drawing experience.

One of the innovations is the two-handed control paradigm. The left hand is the command hand and the right is the brush hand by default. This labor division allows for a sophisticated but easy-to-use set of interactions:

•**State of Drawing:** Close fist with the command hand indicates 'Draw' mode. The software then follows the index finger of the brush hand to draw strokes on the canvas.

•**Selection State**: Extending an open palm on the command hand triggers 'Select' mode. The user can then hold their brush hand's index finger stationary over UI palettes to switch colors or brush sizes.

•**Forced Shape Gestures:** The command hand can shape individual fingers into certain finger arrangements to compel the correction of the subsequent stroke into a specific form. For instance, creating an 'OK' sign preps the system to correct the subsequent drawing into a flawless circle. This provides the user with direct control of the AI aid.

3.3. Automated Shape and Letter Correction

The system's intelligent brain is its correction engine, which works in two basic modes: shapes and letters.

•**Shape Detection & Correction:** If the stroke is finalized and shape correction is turned on, the system employs OpenCV's contour analysis. The stroke is considered as a list of points and is approximated to a polygon by the cv2.approxPolyDP function. The count of the vertices in the resulting polygon is employed to identify the shape (e.g., 3 vertices = triangle, 4 = square/rectangle). For rounded shapes, the system tries to fit an ellipse; a large minor-to-major axis ratio suggests a circle. With the shape identified, the engine creates an ideal version off the original stroke's bounding box and orientation, substituting the user's shaky drawing for a sharp, geometric substitute.

•**Letter Recognition:** In letter mode, the engine takes a different tack. The user's stroke is first translated into a normalized bitmap image. Hu Moments—a group of seven image moments that are translation, scale, and rotation invariants—are computed from this bitmap. These moments are compared against a pre-computed library of Hu Moments for each letter of the alphabet. The closest matching letter is located and painted neatly onto the canvas in place of the original freehand scribble.

3.4. Working Procedure of Futuristic Digital Art

The process from a gesture of the hand to computer art is a fast, live process.

It starts when the webcam records the user's hands and the MediaPipe model recognizes 21 important landmarks in order to construct a skeletal hand model. There is then a Gesture Analysis module that interprets the "command hand's" gestures, like the formation of a fist, to change the system into 'Draw' mode.

In this mode, the system follows the trajectory of the "brush hand's" index finger to create a stroke. This raw stroke is then routed directly into a Correction Engine, which examines its geometry, anticipates the intended shape or letter, and creates a flawless rendition. The final, corrected art is then displayed at once on the canvas, providing an unbroken and interactive creative experience.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁝ Peer-reviewed & Refereed journal ⁝ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141218

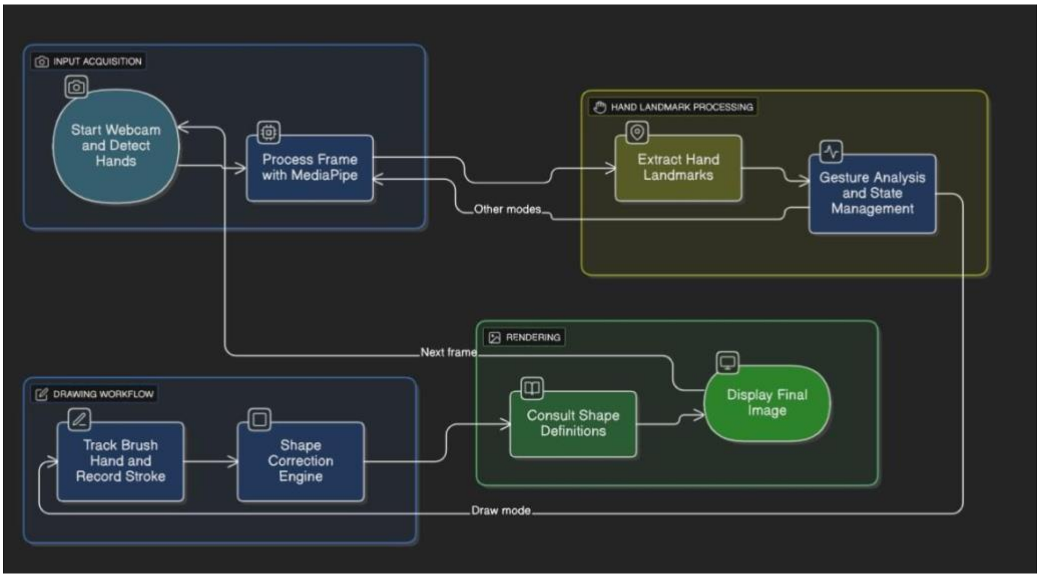

Fig. 3.1 Work flow for Virtual Painter

This flowchart represents the four central modules of the real-time pipeline of AI Virtual Painter.

1. **Input Acquisition:** The webcam is utilized in this first stage to take a live video feed. The frame is immediately passed to MediaPipe model to analyze, as it is the raw material for the entire system.

2. **Hand Landmark Processing:** The MediaPipe model processes the video frame to obtain 21 major landmarks, forming a digital skeleton of the user hand. This organized information is then sent to the "Gesture Analysis" logic, which analyzes the pose of the hand to decide the user's present intent, i.e., going to "Draw mode" or "Select mode."

3. **Draw Workflow:** This module is enabled only if the system is in "Draw mode." It follows the index finger of the "brush hand" to capture the trajectory of a stroke. The finished stroke is then passed to the "Shape Correction Engine," which processes the path intelligently, recognizes the intended geometric form, and computes an ideal version of it.

4. **Rendering:** On the last step, the system renders the completed shape on the digital canvas. This canvas, together with the live view from the camera and UI components, is rendered as the last image presented to the user. The whole cycle recurs instantly for the following frame, providing a smooth and interactive process.

IV. RESULT AND ANALYSIS

The main outcome of this project is a working, real-time AI Virtual Painter application that correctly interprets hand gestures into intelligent digital drawings. The output of the system is an interactive and dynamic user interface with good feedback and expressive drawing features.

The visual output, displayed in Fig. 4.1, includes the following major components:

- A live video feed that is used as the background.
- A hand-tracking overlay that displays the 21 landmarks recognized on the hand(s) of the user in real-time.
- A User Interface (UI) with a color palette, brush size pickers, and mode indicators for the current mode (e.g., Shape-correct: ON, Gesture shape: Rectangle).
- A Gesture Guide that serves as a convenient reminder of gestures used to draw particular shapes.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141218

4.1. Analysis of Performance and Functionality:

The performance and fundamental functionality of the application were tested through real-time interaction, as illustrated in the project video.

1. **Real-Time Performance**: The system exhibits strong real-time performance, running normally between 20-30 FPS on average consumer-grade hardware. This provides for a responsive and smooth user experience, where the virtual brushstroke seems to keep up with the user's finger with little delay.

2. **Gesture Control Accuracy:** The two-handed control system was very accurate. The app consistently differentiated between "Drawing Mode" (fist) and "Selection Mode" (open hand) to enable smooth transitions between drawing and tool choosing. These particular shape-forcing gestures (e.g., raising 5 fingers to choose "Rectangle") were also consistently identified.

3. **State Management & Gesture Analysis (Main Process Step - Blue Box):** The landmark data is now analyzed to interpret the intent of the user. The system inspects the state of the "command hand" (closed fist for 'Draw' mode, open palm for 'Select' mode). The state-dependent logic determines what the movements of the "brush hand" should be interpreted as.

4. **Trace Brush Hand & Store Stroke (Main Process Step - Blue Box):** While in 'Draw' mode, if the system is tracing the path of the index brush hand finger, it records a sequence of (x, y) points that constitute a raw stroke.

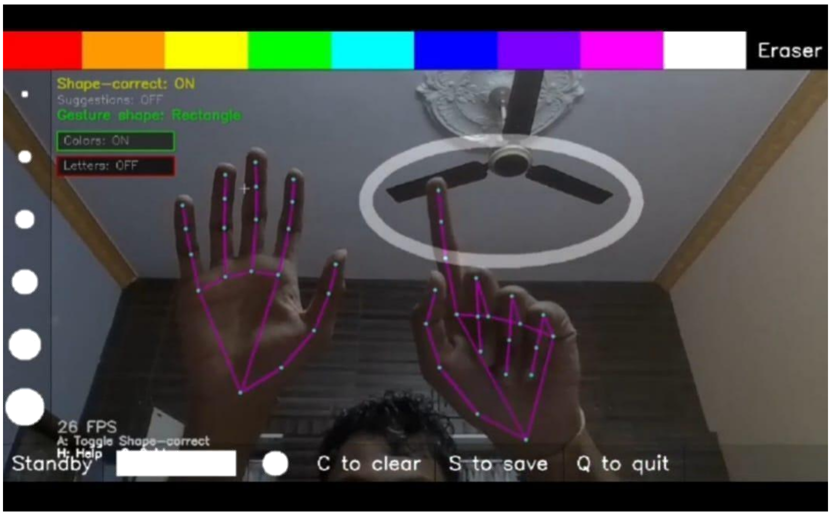

Fig. 4.1 Options display

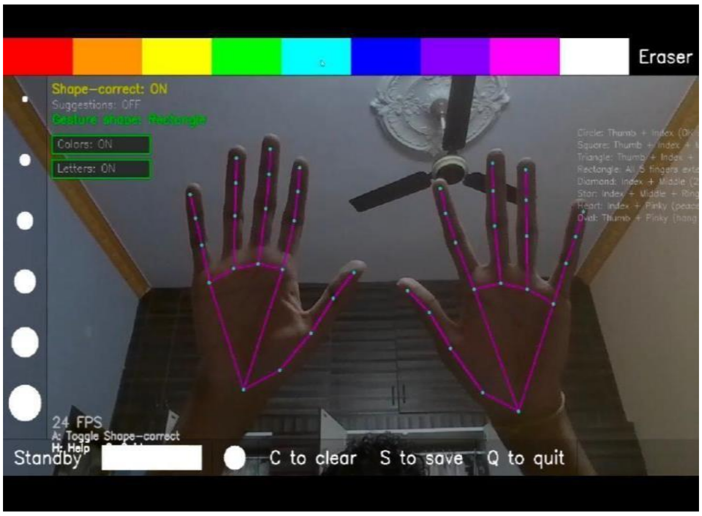

Fig. 4.2 Auto Shape Correct

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⪤ Peer-reviewed & Refereed journal ⪤ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.141218

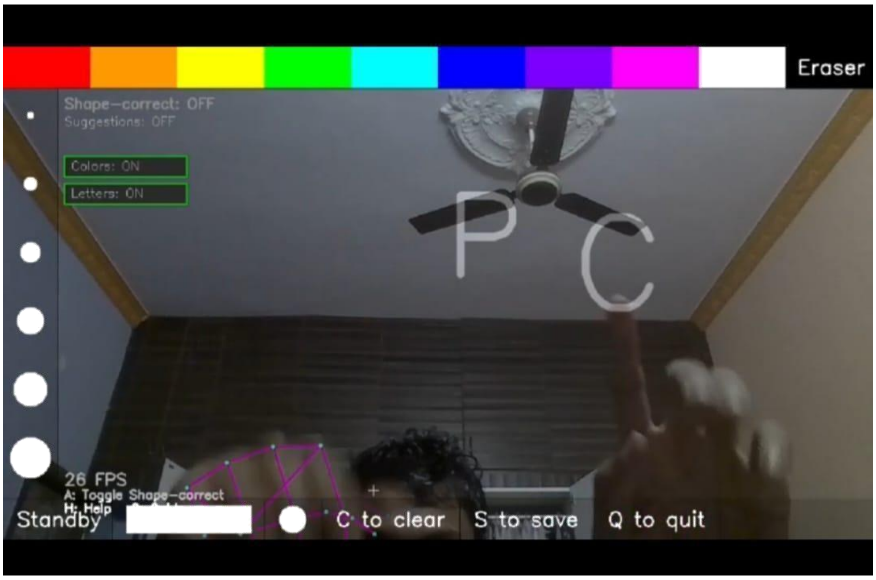

Fig. 4.3 Auto Alphabet Correct

5. **Shape Correction Engine (First Process Step – Blue Box):** After completion of the stroke, it is sent to the correction engine. It examines the geometric characteristics of the stroke with OpenCV's contour approximation for the identification of the user's intended shape.

6. **Refer to Shape Definitions (Database/Knowledge Hub - Green Box):** Once the shape has been forecasted (e.g., "Triangle"), the system refers to its internal knowledge repository, which holds mathematical definitions for how to create a perfect representation of that shape.

7. **Output Rendering to Canvas (Primary Process Step - Blue Box):** The completed shape (or initial raw stroke, if correction is disabled) is then output onto the primary digital canvas**.**

8. **Display Final Image (Final Output - Green Box):** The final composite result is displayed to the user by the system, including the camera feed, the UI elements, and the digital canvas with the drawing. The whole loop is repeated for each frame, creating a smooth and interactive experience.

V. CONCLUSION

"It effectively proves the feasibility of an advanced, gesture-based digital art program that utilizes AI for smart help without the need for hardware with specialized capabilities. Through the careful construction of a two-handed interaction model and an efficient correction engine for figures and characters, the “Futuristic Digital Art” system drastically reduces the entry point to digital creativity. It successfully combines the intuitive expressiveness of hand movements with the accuracy of computational operations.

It is distinguished by emphasizing enrichment of the human creative process, not replacement of it. It is an intelligent assistant that honors user intent while providing strong features to augment and speed their workflow. This shows not only offers a new platform for artists but also to the wider field of Human-Computer Interaction, demonstrating how today’s computer vision can be employed to build more natural and powerful user interfaces.

REFERENCES

[1] "GestureCanvas: An Interactive Web-Based Virtual Whiteboard Using Hand Gestures for Collaborative E-Learning" by P. Sharma and R. Gupta, published in the *2024 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM)*, February 2024.

[2] "Advancements in Gesture Recognition for Digital Art Creation using Mediapipe and OpenCV" by A. Kumar, S. Singh, and M. Verma, published in the *Journal of Artificial Intelligence and Computer Vision*, vol. 3, no. 4, December 2023.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141218

[3] "AirDraw: A Real-Time Hand Gesture-Based Drawing System for Augmented Reality" by S. Kim and J. Park, published in the *Proceedings of the 2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, October 2024 (projected publication).

[4] "GAN-assisted Sketch-to-Painting: A Virtual Tool for Artists using Hand-drawn Inputs" by M. Z. Al-Qurishi, A. F. Al-Badawi, and S. M. R. Islam, published in *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 20, no. 1, January 2024.

[5] "Lightweight CNN Model for Efficient Hand Gesture Recognition in Mobile Virtual Painting Applications" by B. Reddy, P. Kumari, and A. Singh, published in *IEEE Access*, vol. 12, April 2024.Virtual Paint and Volume Control using Hand Gesture by V Shiva Narayana Reddy, K.Kavya, D Sudheer Reddy Geethanjli College of Engineering vol:6 153-159,2022.

[6] A review of Visual Art Using Python and OpenCV by Deepthi B Kuravatti, Deepthi V B , Kulsum N, M Darshan, Sahana MS Vidhya Vardahak College of Engineering published on IRJET e- ISSN:2395-0056 vol-9:May 2022.