Air Writing Gesture Recognition with OCR

Ananda R Kumar Mukkala[1], Muddam Grishmi Reddy[2], Bavirishetty Nithyatej[3], Kalyanam Chinnar[4], Langari Vamshi Krishna[5]

[1]Associate Professor, Dept of CSE, Sreyas Institute of Engineering and Technology
[2]Ug scholar, Sreyas Institute of Engineering and Technology
[3]Ug scholar, Sreyas Institute of Engineering and Technology
[4]Ug scholar, Sreyas Institute of Engineering and Technology
[5]Ug scholar, Sreyas Institute of Engineering and Technology
Corresponding Author: - Ananda R Kumar Mukkala
Associate professor, Dept of CSE, Sreyas Institute of Engineering and Technology

Abstract: **This work presents a new non-communicative technique that mediates atmospheric writing and drawing through gesture recognition. In this system we can draw by capturing the gesture of hand fingers through OpenCV using webcam. The system uses MediaPipe Hands, a state-of-the-art solution for sensing and hand movements. Users can interact with the virtual whiteboard to draw, write or write by moving their hands in the air, and their movements are captured by the web. Use this to choose colors, adjust brush size and switch between pen and eraser types. The smart layer system dynamically displays a series of interactive buttons for additional functions such as hiding or showing the whiteboard, clearing the canvas and customizing drawing settings. These features create a flexible and easy-to-use environment for a variety of applications. This feature continuously monitors the virtual whiteboard, extracts text from the drawing content and converts it to digital text in real time. The text is displayed on the screen and dynamically stored in the file for later use, making it suitable for tasks such as writing, content digitization and collaborative editing. Combining a live webcam with an interactive art canvas. The combination of gesture recognition and OCR shows the potential to create new solutions in education, digital art, remote collaboration and assistive technology. This non-verbal communication is a step towards optimal human-computer interaction, providing a variety of platforms for future development and implementation.**

Keywords—Gesture-Recognition, OpenCV, MediaPipe, Optical Character Recognition (OCR).

I. Introduction

This work combines real-time experience with computer vision to offer a flexible and engaging way to write and draw in the air. The system leverages Python libraries such as OpenCV, MediaPipe Hands, and NumPy for tracking and movement, allowing users to manipulate and interact with virtual data seamlessly and with understanding. With this setup, users can create drawings, write notes, and edit tools without physical contact. The system facilitates an interactive experience where users can manipulate the canvas' functions through gesture recognition, making the canvas adaptable to a variety of creative and educational activities. Brush size changes, transitions between standard whiteboard or custom canvas layouts, and more. The clear and effective eraser mode allows users to erases text, while the ability to erase the entire canvas provides full control over the workspace. These features increase the usability of the system, making it suitable for a variety of applications, from art and design to education and presentation.

Number format. These features enhance the system's capabilities and make it easier for users to capture and send written content. Whether used for text, graphics, or educational programs, OCR preserves the value of user input by ensuring that all text is accurate and efficient. It is safe and effective, providing a dust-free and environmentally friendly alternative to traditional blackboards and plastic markers. This makes it especially useful in environments where reducing waste is important, such as classrooms and online courses. The interactive nature of this system provides a virtual whiteboard experience that could revolutionize digital education by facilitating participation in e-learning. Combining artificial intelligence, computer vision and motion control, the project demonstrates the technology's potential to create new, sustainable learning tools.

II. Objectives and Methodology

This project aims to revolutionize how users interact with digital content by providing new atmospheric text and graphics. It uses new technologies such as OpenCV, MediaPipe and in-plane character recognition (OCR) to provide a contactless, intuitive way to draw, write and interact with virtual files. Thanks to real-time workflows, users can use simple gestures to manipulate data to create and edit content seamlessly and intuitively. The simplicity and ease of use of the system make it an ideal tool for many applications, especially in creative and educational environments. Touch image processing and gesture recognition have been improved. Head tracking and gestures work. Make sure the pointer is accurate even in a good environment, let users interact well. writing. To improve usability.

III. Literature Survey

Much of the research carried out in the field of gesture recognition focuses on general-purpose applications such as gaming, virtual reality, and sign language interpretation. However, systems like the air canvas that combine gesture recognition and Optical Character Recognition (OCR) for real-time applications are relatively underexplored. These systems leverage computer vision and advanced interaction technologies to allow users to interact with devices through hand gestures, offering a seamless and intuitive user experience. Despite notable advancements, the practical utility of such systems is limited by recognition accuracy, environmental constraints, and real-time processing challenges.

Gesture recognition is at the core of interactive systems like the air canvas. MediaPipe, a widely used framework for hand tracking, has been instrumental in enabling real-time gesture detection and tracking. Recent studies have demonstrated the effectiveness of MediaPipe in capturing dynamic gestures for various applications. However, challenges such as occlusions, varying lighting conditions, and complex hand movements still affect the robustness of gesture recognition systems.

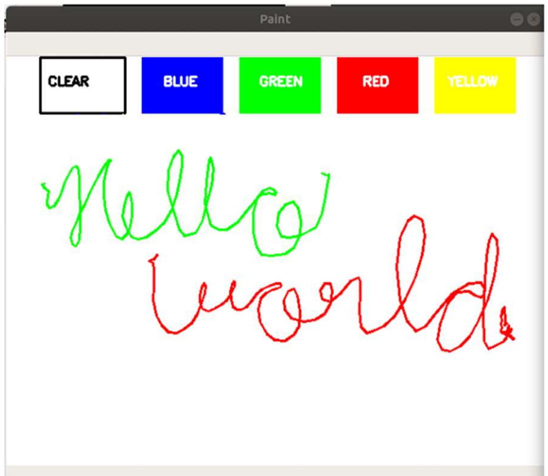


Fig. 1: Gesture Recognition using MediaPipe

OCR plays a crucial role in systems that aim to recognize and process air-written text. The implementation of Tesseract OCR, an open-source engine, has enabled the conversion of hand-drawn gestures into readable text in real time. Research has shown that preprocessing steps like binarization and grayscale conversion enhance OCR accuracy significantly. While OCR integration improves the usability of gesture-based systems, its accuracy is influenced by the quality of input gestures and preprocessing techniques.

Fig. 2: Integration of OCR to text recognition

IV. Proposed System

The proposed system facilitates interactive work with virtual documents and tasks such as drawing, erasing, color and brush size selection, and real-time optical text recognition (OCR). This system integrates advanced computer vision technologies, including MediaPipe for hand tracking and Tesseract for OCR, to create a user-friendly, gesture-controlled interface. The following is a detailed and structured description of the system:

A. Hand Tracking with MediaPipe:

To track and process hand gestures for controlling the virtual canvas and interacting with user interface (UI) elements.
Leverage MediaPipe Hands, a powerful tool that detects and tracks 21 hand landmarks. Each webcam frame is processed to detect the hand and capture the movement of the index finger tip, which acts as the primary tool for interaction.

Key Features:
Gesture Mapping: Map the index finger's coordinates to specific actions on the canvas, such as drawing, erasing, or selecting buttons.
High Precision: MediaPipe provides accurate tracking for all 21 hand landmarks, ensuring smooth and responsive interaction.
Actions: Enable specific gestures (e.g., pointing, hovering) to trigger interactions such as drawing, clearing, or toggling visibility of UI elements.

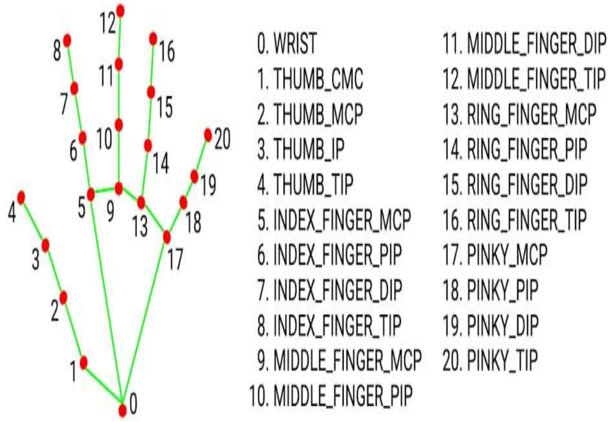


Fig. 3: 21 hand landmarks

B. Interactive User Interface (UI) for Canvas Control:

The goal is to design a user-friendly interface for managing a virtual canvas represented as a NumPy array, enabling users to draw and interact with it effectively. The interface will feature a blank virtual canvas that records user interactions and interactive control buttons for selecting colors, adjusting brush sizes, and toggling the visibility of the whiteboard and other UI elements. A ColorRect class will manage UI elements like color palettes, brush size selectors, and control buttons, rendered with OpenCV functions like cv2.rectangle and cv2.putText.

Users can choose from a predefined color palette, adjust brush sizes for precision or bold strokes, and activate an eraser using black or a dedicated button. A "Clear" button will allow the canvas to be wiped clean effortlessly. Additionally, the system provides a gesture-controlled interface for seamless interaction. Key actions include showing/hiding color palettes with a "Color Button," toggling brush size options with a "Brush Size Button," and controlling whiteboard visibility with a "Whiteboard Button." The interaction logic ensures intuitive control, activating functions when the index finger hovers over a button. This design promotes an engaging, efficient, and hands-free experience for managing the virtual canvas. The interface also incorporates visual feedback to enhance usability, displaying selected options or active tools on the screen. These features, combined with gesture-based controls, ensure an immersive and efficient drawing experience tailored to user preferences.

Fig. 4: User Interface

C. Real-Time Optical Character Recognition (OCR):

To enable real-time recognition and processing of handwritten or drawn text on the virtual canvas, the implementation involves continuous monitoring of the whiteboard area for text recognition. Preprocessing steps include converting the canvas image to grayscale for uniformity and applying binarization to enhance text visibility for OCR. Tesseract OCR, accessed via pytesseract.image_to_string, is used to extract text from the processed image. The system features continuous OCR for real-time text recognition, dynamic text saving to a file (e.g., recognized_texts.txt) for future reference, and visual feedback by displaying the recognized text near the "OCR" button on the UI. This functionality ensures an interactive and efficient approach to managing and preserving handwritten or drawn content.

D. System Workflow:

The workflow begins with the initialization phase, where the webcam capture is started, and MediaPipe Hands is initialized for gesture tracking. In the main loop, each video frame is processed to detect hand gestures and map them to corresponding actions on the canvas or UI, such as drawing, toggling options, or interacting with the whiteboard. Simultaneously, OCR is performed on the whiteboard area to recognize and save text dynamically, with the recognized text displayed on the screen. The system continues this process until the user presses the "q" key, which serves as the exit condition, terminating the application.

Fig. 5: Real-time recognition of text

V. IMPLEMENTATION

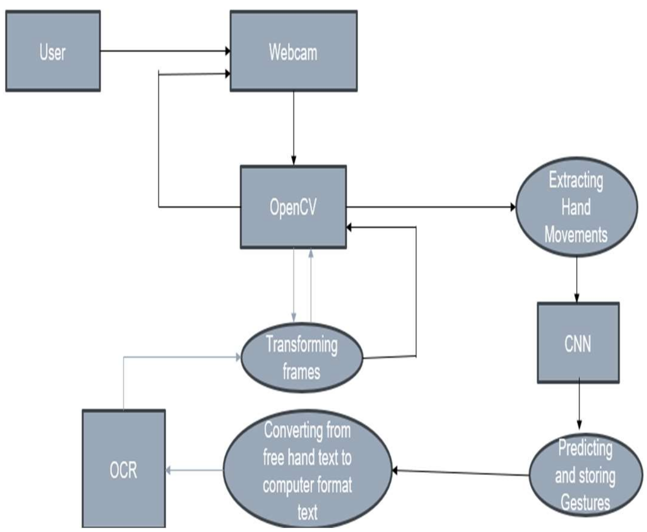
The webcam feed is continuously captured and processed using MediaPipe to detect hand landmarks (particularly the position of the index finger).

The index finger's position is used to draw on the canvas. The cv2.line() function is used to draw the lines based on the user's movement. The color and brush size can be selected using the buttons displayed on the screen. If the user chooses the "Eraser" color, it draws lines with the eraser size to remove previous drawings.

The text on the canvas is recognized using pytesseract.image_to_string() and displayed on the screen. The recognized text is written to a text file (recognized_texts.txt) for future reference.

The code uses ColorRect class to create clickable buttons for actions like changing the pen size, selecting a color, clearing the canvas, and toggling visibility of the drawing board.

A. Architecture Diagram:



The system starts with the user **giving** input **via** gestures or **text recorded** by **the** webcam. OpenCV processes the video frames to extract and **convert** them for analysis. Hand movements are tracked and analyzed using gesture recognition **tools, and t he movements are predicted and stored as preorders using** a **convolutional neural network (CNN). Meanwhile,** Optical Cha racter Recognition (OCR) **performs frame conversion** to **identify text** and convert **it to a** computer format.

The **combination** of gesture recognition and text recognition **allows** the system to provide an intuitive **way to interact** w ith digital **data, enabling** interaction **via** gestures and **text.** Define a ColorRect class for creating UI elements like color palettes, brush size selectors, and control buttons.

B. Test Cases:

Test Case Id	**Scenario**	**Steps**	**Expected Output**	**Actual Output**	Status
TC1	Hand Detection	Launch the application	Hand landmarks	Hand detected and	Pass

		and move your hand within the webcam frame.	are detected and tracked in real-time.	tracked successfully.	
TC2	Color Selection	Select a color from the palette and draw on the canvas.	The selected color is used to draw lines on the canvas.	Lines appeared in the selected colors.	Pass
TC3	Brush Size Selection	Choose a brush size and draw lines.	Lines are drawn with the selected thickness.	Brush size worked correctly.	Pass
TC4	Whiteboard Visibility Toggle	Click the Board button to show or hide the whiteboard.	Whiteboard visibility toggles as expected.	Toggle worked correctly.	Pass
TC5	Click Outside Button Area	Click randomly outside the button areas.	No action is triggered, and the application runs normally.	No unexpected actions occurred.	Pass
TC6	Multiple Hand Detection	Launch the application and move your hand within the webcam frame.	Hand landmarks are detected and tracked in real-time.	Hand detection and track unsuccessful.	Fail
TC7	Exit Application	Press the 'q' key to quit the application.	The application closes without errors.	Application exited successfully.	Pass
TC8	Clear Canvas While Drawing	Attempt to click the Clear button while drawing on the canvas	Canvas is cleared, and the application continues without error.	Canvas cleared successfully during drawing	Pass

TC9	Converting free hand text to digital format	Write on the canvas	Dynamically convert free hand text to digital format	Dynamically converted free hand to digital format	Pass

VI. Discussion

A. Comparative Analysis:

The code provides new solutions for real-time image orientation and optical character recognition (OCR), providing interactive, feed-free Customer management, making it different from the traditional process. Unlike traditional drawing software that relies on physical tools such as pen, pencil or mouse, the system interacts with gestures, allowing users to manipulate the material without the material. This eliminates the dependence on physical equipment, making the body versatile and practical. Similarly, most existing OCR methods require static images or pre-written texts to be recognized without real-time interaction with digital graphics. The use of MediaPipe Hands for gesture recognition and Tesseract OCR for instant text, art, and text recognition makes it unique compared to graphics or OCR software. Software like Adobe Photoshop or CorelDRAW offer better editing and advanced features like a variety of brushes, decorative tools, and better integration with other created tools. These features give professionals more flexibility in their work and the ability to perform multiple design tasks. In comparison, existing methods are limited in terms of brush options and drawing methods; they focus on simplicity and instant interaction rather than providing advanced customization.

It usually works well for clear and beautiful texts, but can be difficult in complex cases. For example, background noise, poor handwriting, or texts with multiple characters can negatively affect the accuracy of the OCR. This is a limitation of OCR systems, and this application is no exception. The accuracy of the system depends on the quality of the captured points. Although Tesseract OCR provides good verification results, it may not be able to recognize poorly written or poorly typed text, which may affect the reliability of the system. A clear view of the userâ s hands, which can be challenging in environments where the userâ s hands are obstructed or not visible at all. The system will be accessible in crowded places, as users will not have a clear view to track their movements. This limitation may limit its effectiveness in dynamic or difficult environments, such as a classroom with many people or a presentation where the userâ s hands are not completely within the camera. The limitation is that the current system only supports one user at a time. The application can track the movements of one user at a time, which can be a disadvantage in a collaborative system where many people may want to interact with the screen at the same time. Extending this functionality to support multiple users or multi-handed tracking would be useful in group work such as interactive sessions or collaborative design.

B. Positive Aspects:

One of the best features of this system is its real-time chat. Users can draw, erase, and manipulate data clearly by pointing, creating a hands-free environment that enhances user experience and engagement. MediaPipe Hands plays a key role in this feature, providing accurate and responsive hand tracking. This eliminates the need for traditional devices such as a mouse or pen, which can be difficult or less intuitive in some applications. In one study, for example, this allowed teachers to bring a pen or mouse to learn information without interrupting instruction.

The Tesseract OCR component adds significant value by providing instant reading of the canvas in addition to the drawing guide. When the user draws or writes, the system instantly recognizes the drawing and converts it into a readable form. The ability to capture and analyze writing is now essential in events such as academic presentations, conferences or live presentations. These features, which are not common in many drawing tools or annotations, allow this system to stand out over time from the combination of drawing and text recognition. Size, color selection and drawing type (pencil or eraser). This flexibility allows the system to adapt to different settings, whether it is used for drawing, typing or chatting. Users can switch between different colors and sizes to ensure their drawings and annotations are clear and distinct. This change is especially important in situations that require fast, intuitive interactions, such as presentations or collaborative discussions where speed and clarity are important. The future of its application can be expanded to multiple user environments. The system currently supports single-user viewing, but the user experience can be expanded to support multiple users or hands. Adding this functionality will make the system more versatile and more collaborative. The ability to manage multiple users in spaces like classrooms, training sessions, or brainstorming sessions will encourage team collaboration and encourage collaboration and engagement. Additionally, it can be adapted for virtual meetings where participants can interact with shared content from different locations or devices, providing a new way for remote teams to collaborate. This method is useful for learning and analysis. As users write or draw, the system can instantly recognize and process the text, allowing for immediate review and correction. This feature is especially useful in educational environments where teachers or students can get instant feedback on text or graphics, reinforcing learning and understanding, hands-free interaction is ideal for scenarios where physical input devices are impractical or unnecessary. The combination of tracking and OCR information now provides users with new and engaging interactions with digital content, creating new use cases in education, design, and collaboration

VII. CONCLUSION AND FUTURE SCOPE

This project achieves the integration of computer vision technology by combining MediaPipe Hands for hand tracking, OpenCV for realistic drawing, and Tesseract OCR for optical character recognition. The application provides a great virtual experience that allows users to use gestures to draw on a digital canvas, select colors, adjust brush size, and erase a section of the canvas. It can also recognize text on the canvas and offers an interactive way to perform OCR. The combination of these features demonstrates the terminal's functionality and its potential for practical use in education, art, and production tools. Enhanced possibilities. First, you can add advanced gestures that allow gestures to perform tasks such as zooming, resizing, and undo/redo operations. Expand the program to support multi-hand tracking for hands-on coordination or multi-finger gestures for greater interactivity. Increasing OCR accuracy through advanced optimization techniques (like noise reduction and transformation) and adding support for multiple languages can make the tool even more powerful. Allow users to save their work or send approvals to cloud storage platforms like Google Drive or Dropbox. Bringing an app to a mobile or web platform will make it more accessible and reach a wider audience. Finally, incorporating intelligent models to better predict gestures and use better language and text recognition techniques can improve performance and user experience. These future developments could turn the app into a powerful tool for use in a variety of situations, including education, design, and collaboration.

VIII. REFERENCES

[1] G. Tauschek, "Reading machine," U.S. Patent 2 026 329, Dec. 1935. [23] P. W. Handel, "Statistical machine," US. Patent 1915 993, June 1933.

[2] Amma C, Georgi M, Schultz T. Airwriting: Hands-free mobile text input by spotting and continuous recognition of 3D-space handwriting with inertial sensors. In2012 16th International Symposium on Wearable Computers 2012 Jun 18 (pp. 52-59). IEEE.

[3] Shalini M, Indira B. Automatic Character Recognition of Indian Languages–A brief Survey. International Journal of Innovative Science, Engineering & Technology. 2014 Apr;1(2):131-8.

[4] Hsieh CH, Lo YS, Chen JY, Tang SK. Air-writing recognition based on deep convolutional neural networks. IEEE Access. 2021 Oct 21;9:142827-36.

[5] Ahmed S, Kim W, Park J, Cho SH. Radar-Based Air-Writing Gesture Recognition Using a Novel Multistream CNN Approach. IEEE Internet of Things Journal. 2022 Jul 8;9(23):23869-80.

[6] Wang C, Liu J, Chen Y, Liu H, Xie L, Wang W, He B, Lu S. Multi-touch in the air: Device-free finger tracking and gesture recognition via cots rfid. InIEEE INFOCOM 2018-IEEE conference on computer communications 2018 Apr 16 (pp. 1691-1699). IEEE.

[7] Anjaneyulu P, Jampaiah Y, Karthik R, Vijetha T. Air writing recognition modeling and recognition of characters, words and connecting motions. In2017 International Conference on Intelligent Sustainable Systems (ICISS) 2017 Dec 7 (pp. 1112-1115). IEEE

[8] Kumar P, Saini R, Behera SK, Dogra DP, Roy PP. Real-time recognition of sign language gestures and air-writing using leap motion. In2017 Fifteenth IAPR international conference on machine vision applications (MVA) 2017 May 8 (pp. 157-160). IEEE.

[9] Tsai TH, Hsieh JW, Chen HC, Huang SC. Reverse time ordered stroke context for air-writing recognition. In2017 10th International Conference on Ubi-media Computing and Workshops (Ubi-Media) 2017 Aug 1 (pp. 1-6). IEEE.

[10] Tan X, Tong J, Matsumaru T, Dutta V, He X. An End-to-End Air Writing Recognition Method Based on Transformer. IEEE Access. 2023 Oct 4.

[11] Zhang H, Chen L, Zhang Y, Hu R, He C, Tan Y, Zhang J. A wearable real-time character recognition system based on edge computing-enabled deep learning for air-writing. Journal of Sensors. 2022 May 4;2022.

[12] Ye Z, Zhang X, Jin L, Feng Z, Xu S. Finger-writing-in-the-air system using Kinect sensor. In2013 IEEE international conference on multimedia and expo workshops (ICMEW) 2013 Jul 15 (pp. 1-4). IEEE.

[13] Convert Handwriting to Text – Transkribus AI Software. https://www.transkribus.org/handwriting-to-text

[14] 15 Best Handwriting to Text Converter Apps – TechCult.com [Online]. Available: 15 Best Handwriting to Text Converter Apps 2024 – TechCult

[15] How to convert handwriting to text with OCR to PDF [Online]. Available: https://www.adobe.com/acrobat/hub/useocr-to-turn-handwritten-text-into-pdf-files.html

[16] Handwriting Recognition and Conversion Using Neural Networks, https://link.springer.com/chapter/10.1007/978-981- 19-5224-1_16

[17] Liu, Brian & Xu, Xianchao & Zhang, Yu. (2020). Offline Handwritten Chinese Text Recognition with Convolutional Neural Networks. 10.48550/arXiv.2006.15619.

[18] R. Malhotra and M. T. Addis, "End-to-End Historical Handwritten Ethiopic Text Recognition Using Deep Learning," in IEEE Access, vol. 11, pp. 99535-99545, 2023, doi: 10.1109/ACCESS.2023.3314334.

[19] Ruchika Malhotra, "Handwritten Amharic Word Recognition Using Additive Attention Mechanism," IEEE Trans. Multimedia, vol. 21, no. 1, pp. 221–233, Jan. 2024.

[20] Y. Wang, Y. Yang, W. Ding and S. Li, "A Residual-Attention Offline Handwritten Chinese Text Recognition Based on Fully Convolutional Neural Networks," in IEEE Access, vol. 9, pp. 132301-132310, 2021, doi: 10.1109/ACCESS.2021.3115606.