

Blind Peer Reviewed
(UGC-CARE)
ISSN: 0972-2750

Manuscript Received on –
30th June, 2023

Primary Reviewed on –
6th July, 2023

Peer Reviewed on –
24th July, 2023

Accepted for publication
1st October, 2023



Dr. Geeta Padole
Assistant Professor
Department of CSE
Gokaraju Rangaraju Institute of
Engineering and Technology
Hyderabad
geeta1671@grietcollege.com

Bashpangu Yuvaraj
Dept. of Computer Science Engineering
Gokaraju Rangaraju Institute of
Engineering and Technology
Hyderabad, India
yuvarajbashipangu@gmail.com

Arjun Shettigar
Dept. of Computer Science Engineering
Gokaraju Rangaraju Institute of
Engineering and Technology
Hyderabad, India
arjun.shettigar16@gmail.com

Mudinenni Mokshith Naidu
Dept. of Computer Science Engineering
Gokaraju Rangaraju Institute of
Engineering and Technology
Hyderabad, India
mokshithnaidu@hotmail.com

Potu Aryavardhan
Dept. of Computer Science Engineering
Gokaraju Rangaraju Institute of
Engineering and Technology
Hyderabad, India
aryavardhan.potu@gmail.com

Gesture-Controlled Virtual Drawing Board with Voice Command Integration

Abstract- In the present-day digital world, there is growing need for interactive environments in which users can draw and engage with virtual space by using natural motions without any need for physical interaction with devices. Gesture recognition is proving to be a valuable capability that allows effortless user interactions while voice command integration makes it all hands-free friendly. The present project showcases a virtual blackboard system that utilizes hand gesture recognition and voice input to deliver an immersive and touch-free drawing experience. Implemented with Python, OpenCV, and MediaPipe, the system recognizes hand movement in real time, enabling users to draw electronically using easy gestures. Voice inputs can also be used as commands for functions, allowing users to erase the board or save their work easily. Through the application of computer vision techniques, a key part of artificial intelligence, the system reads images from a webcam to monitor and process hand locations. Even though many virtual drawing tools are available, the integration of voice control makes this project stand out, since it allows the project to be more interactive and user-friendly. The output of this project is expected to be an optimal, effective and a user-friendly virtual blackboard that accurately recognizes hand gestures and voice commands with reduced latency. This technology finds uses in education, digital art, and accessibility tools, providing an easy method for users to engage with digital platforms.

Keywords- Virtual Drawing Board, Hand Gestures, Computer Vision, OpenCV, MediaPipe, Voice Command

I. INTRODUCTION

This Virtual Drawing Board is a contactless digital painting system with hand gesture and voice-controlled creative interaction. This work investigates the ability of contactless technology to enable natural and intuitive digital painting with natural human movements. Today's tools require physical contact with dedicated devices, restricting creativity. In contrast, the Virtual Drawing Board converts a

standard webcam into an instant drawing interface with computer vision and gesture identification [1], [2].

The system interprets hand movement and finger position into drawing commands so that users can draw and switch tools without restrictions. This is made possible based on groundbreaking work in computer vision that has grown tremendously in the last ten years [4], [5]. The system exploits deep convolutional neural networks to recognize hand shapes,

which have proved incredibly accurate under laboratory settings [6]. Freehand drawing is enabled but so is object modeling and 3D modeling, with no vast amount of experience necessary for classic drawing programs.

in education and in fine arts, through the elimination of real-time obstacles to on-line creation [7], [9].

II. RELATED WORK

A number of researchers have also been engaged in the design of gesture-based virtual drawing systems. Gayathri Mumali et al. (2023) designed an AI-Based Virtual Drawing System using OpenCV and MediaPipe to track 21 hand landmarks for touchless drawing. Their system performed quite efficiently in giving various modes of drawing, like shapes, freehand, and colour [16]. Narasimha et al. (2024) developed a "Whiteboard Application using Machine Learning" model that solved accessibility issues brought about by expensive touch devices. Their application based on MediaPipe allowed users to participate in drawing activities through different hand gestures that were useful in distinguishing selection and drawing modes [17]. Pranavi Srungavarapu et al. (2021) developed "Virtual Sketch using OpenCV," which used colour markers and advanced image processing methods such as morphological processes to develop a natural interface in which the hand movements get converted into digital sketches [1]. Faria Soroni (2021) developed a "Hand Gesture Based Virtual Blackboard" only for the purpose of online education during the COVID-19 pandemic. The system achieved a recording accuracy of approximately 90% in detecting handwritten material sketched in the air [18]. Some of the other notable contributions include Rohini Shirkar et al.'s (2023) "Virtual Mouse using Hand Gestures," which achieved a 99% overall accuracy in translating mouse movements from hand gestures [19], and Farah Farhana et al.'s (2019) "Real-Time Hand Gesture Recognition for Embedded System," which was aimed at helping patients with conditions like hyperhidrosis through a low-cost interface [7]. Nogales and Benalcázar (2023) examined deep machine learning techniques and contrasted the use of manual and automated feature extraction processes and concluded that BiLSTM-ANN achieved an

Voice integration provides accessibility that allows for erasure of the board, a screenshot, or modification of brush properties by voice [11], [15]. Physically disabled users are supported through multimodal input as well as enhanced creative participation. Recent developments in speech recognition technology have made real-time voice control ever more reliable, with scope for use extending beyond creative software to industrial automation [11] and robotic control systems [15]. Designed as an internet product, the system is hardware-independent and accessible through any standard web browser, requiring only a webcam. This strategy is consistent with research proving the efficacy of browser-based gesture recognition systems [10], providing entry barriers for users without special equipment. The benefits of accessibility are especially important in educational environments and in circumstances where remote interaction is necessary, as proven during the COVID-19 pandemic [8].

Combining high technical ability with an unelaborate presentation approach places the system within reach of millions, hiding high-level processing under a very friendly interface. The virtual drawing board shows how machine perception and software mimicry could make computer-assisted human-machine conversation transparent, especially

accuracy rate of 99.9912%. Chaur Hsieh et al. (2021) employed an air-writing recognizer based on deep convolutional neural networks with a accuracy of over 99% [12].

Kamlakant Bag et al. (2023) developed a "Virtual Whiteboard-A Gesture Controlled Pen-free Tool " with hand gesture control and presentation assistant [20], while Saoji et al. (2021) developed an "Air Canvas Application" with 99% in the laboratory setting and 94% with changing backgrounds [13]. This research demonstrates that machine learning approaches can achieve high accuracy even in non-ideal environments, supporting the feasibility of our system for everyday use.

Collectively, the research exhibits breathtaking progress toward developing functional, accessible, and accurate hand gesture-based virtual drawing and interactive systems.

III. PROPOSED SYSTEM

The Virtual Drawing Board is a contactless way of drawing with computer vision and artificial intelligence,

converting the user's hand into an ad-hoc input device from a regular webcam. The system can respond to natural hand movements and voice commands, crossing boundaries between human creativity and digital presentation by rendering technology nearly invisible. The system is developed with a Python backend and operates on five independent layers:

Video Acquisition Layer: Records webcam frames and pre-processes images for the MediaPipe Hand Track model.

Gesture Interpretation Layer: Translates 3D hand landmarks to infer finger positions and translate them to particular functions.

Drawing Engine Layer: Manages canvas state and executes drawing operations on interpreted gestures.

Speech Processing Layer: Processes voice commands concurrently with the main program.

Web Presentation Layer: Places the interface and video stream over a browser.

The system's functionality is based on the "fingerUp()" method, which checks whether a finger is open or closed and initializes a Boolean array for the five fingers. This allows the system to identify various modes:

Drawing Mode: Only Index finger up.

Selection Mode: Both Index and middle fingers up.

Every tool translates hand movement in a different way; freehand drawing tracks the index fingertip to render lines, shape tools determine sizes from distances between initial and present finger locations and the eraser tool renders white lines to delete current material. The system architecture, for the gesture detection module and speech recognition is shown in Fig. 1.

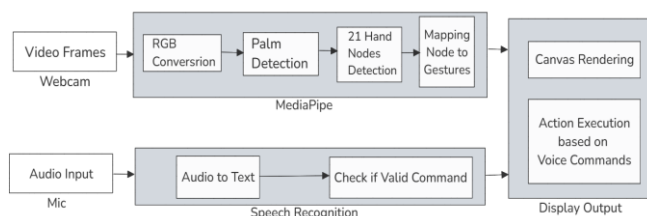


Fig. 1. Proposed System Architecture

IV. LIBRARIES

OpenCV:

It is an open-source library for computer vision which is necessary for capturing webcam input, image processing in real-time, and performing different drawing functions on the canvas. OpenCV manages all image manipulation operations, including color conversions, shape drawing, and bitwise operations required for overlaying drawings onto the live video feed of the user.

MediaPipe:

MediaPipe is a framework offered by Google, utilized for hand node point detection and tracking. It offers hand position information with accuracy, essential for the gesture recognition. The hand tracking algorithm of MediaPipe detects 21 nodes on each hand such that the application is able to trace finger positions and comprehend the gestures used for drawing and for tool selection.

SpeechRecognition:

This library allows the voice command functionality by converting speech input to text output. It interfaces with Google's Speech Recognition API to process audio inputs from the user's microphone and convert them into text command functions that can be understood by the application.

PyAudio:

Links with PortAudio, a cross-platform audio I/O library. PyAudio is used to access the system audio devices and record or capture the microphone input for the voice command system. It allows low-level audio functions necessary for real-time voice recognition.

NumPy:

Utilized for efficient array operations and mathematical computation involved in the application. NumPy is utilized for data structure creation to hold canvas data and to execute operations on pixel data. It's particularly beneficial for operations such as smoothing hand movement and geometric shape calculations.

V. MODULES

Main Application Module:

This module is essential and important for all Virtual Drawing Board functionalities. It initializes main components like the webcam framework, canvas layers and client UI components. This module actualises the essential events that analyses video frames, recognises hand points, and translates movements to execute appropriate drawing activities. It handles the main motion detection logic by analysing hand node points to decide which fingers are expanded, utilizing this data to recognize between drawing mode (index finger extended only) and selection mode (index and middle fingers extended). The module oversees state changes between distinctive tools and combines video input with user's drawings. It arranges interaction between distinctive subsystems, guaranteeing that user inputs, whether through gestures or voice commands are properly interpreted and reflected within the application. Moreover, it actualizes the stroke rendering framework with variable thickness controls and colour selection capabilities. The module moreover handles shape creation instruments like circles, rectangles, triangles, squares, 3d shapes like cubes and cuboids. At last, it oversees the overlay of UI components onto the processed video feed before giving it to the internet interface.

Hand Tracking Module:

The Hand Tracking Module is the main component which is controlling the gesture-based interface, empowering precise and real-time identification of hand node points for the application. Built on Google's MediaPipe platform, it analyses video streams of users to detect hand or palm areas with accuracy. Through analysis of the geometric relations among the recognized 21 landmarks as shown in Fig. 2, the module computes hand orientation and finger configurations, setting the stage for proper gesture interpretation. It is resilient against a variety of hand poses and orientations, providing reliable tracking regardless of how the hand is oriented with respect to the camera. It extracts fingertip coordinates and joint angles, which serve as key inputs for the gesture recognition engine. Furthermore, the system can differentiate between left and right hand, allowing future enhancements that involve coordinated two-hand interactions.

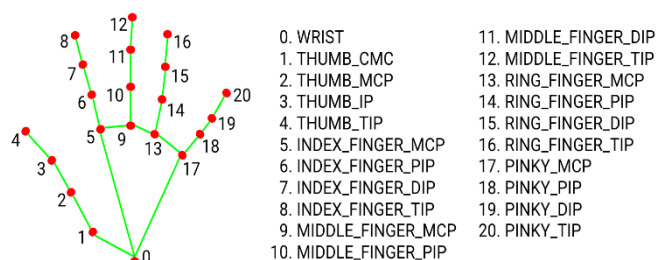


Fig. 2. 21 Hand Landmarks

Voice Command Module:

This Module allows voice-controlled interaction with the virtual drawing board, offering the clients to operate the system without their hands. Designed to work independently, it performs in an isolated thread to ensure the application remains responsive and independent of voice processing operations. The module initializes the audio input system with PyAudio. It can support multiple microphones and switch automatically to a connected device in case the default one is unavailable. To support stable speech recognition in various environments, it pre-calibrates for background noise before listening for input. Audio is recorded continuously in brief bursts and the audio is sent to Google's Speech Recognition API. There it is converted into text. The module uses a bank of pre-defined commands that relate to some app actions, identifying direct sentences like "draw circle" and short forms like just "circle". It also supports smart pattern recognition, i.e. it is able to recognize commands even if not spoken correctly or as part of a complete sentence. Feedback from the module to the user includes showing some temporary markers on the interface whenever a command is recognized. In the event of mishaps such as network or audio hiccups, the module has the capability to restart automatically, thereby providing unbroken and seamless operation. The module also incorporates sophisticated voice commands for functions such as tool changing, brush size, clear canvas and screenshot.

Web Server Module:

The Web Server Module drives the rendering of the application interface with a HTTP server that is executed by employing Flask. It is also responsible for the backend routes that are the skeleton of the application such as the landing page and the drawing canvas. This module can stream live video from OpenCV by changing video frame feed into a browser format using HTTP responses. For better browser performance and compatibility, it sets precise cache control and content-type headers. The module uses templating system of Flask to dynamically generate HTML, injecting required variables and paths for fine rendering. The module analyses user-triggered actions that require server-side logic, such as taking and saving screenshots. It ensures all content is served with the correct MIME types and includes error handling mechanisms to gracefully manage issues like missing files or malformed requests. Due to its web-based architecture, this module allows the drawing board to run on virtually any modern device equipped with a browser and webcam—no extra setup or installations needed. It also lays

the groundwork for future features like real-time collaboration and cloud-based synchronization.

VI. RESULTS

The Virtual Drawing Board project with its user-friendly interface shown in Fig. 3, successfully achieved its goal of creating a contactless drawing experience accessible through standard webcam technology. The system demonstrates reliable hand landmark detection and accurate gesture recognition, allowing users to switch seamlessly between drawing and selection modes.

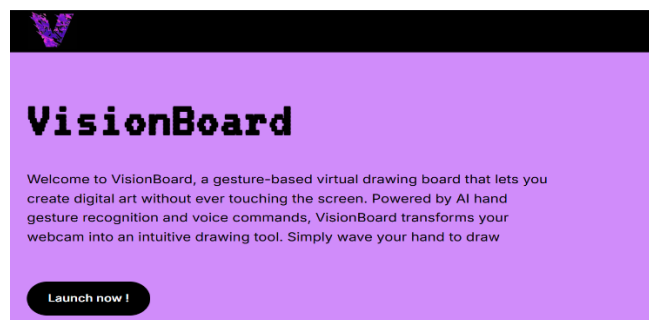


Fig. 3: User Interface

The drawing functionality performs effectively across various tools, including freehand drawing, shapes (rectangle, circle, triangle, square, cube, and cuboid), and eraser functions, and is presented in Fig. 4 and Fig. 5. Colour and eraser selection operates smoothly through the header menu interface.

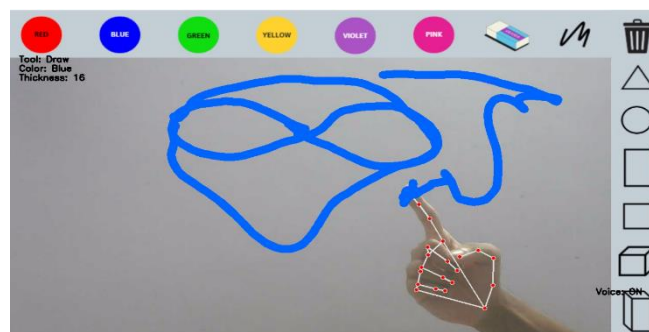


Fig. 4: Free hand drawing

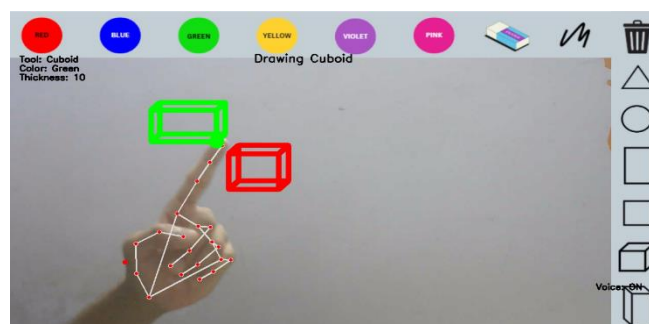


Fig. 5: Shape drawing and Colour Selection

Voice command integration enhances usability, with the system properly responding to commands for tool selection, canvas clearing, and screenshot capture. The application maintains responsive performance even with continuous

gestures and complex drawing operations, and this is illustrated in Fig. 6 and Fig. 7.

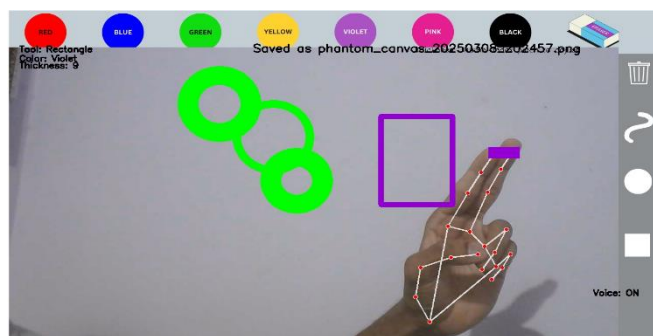


Fig. 6: Shape drawing via Voice Commands



Fig. 7: Thickness control via Voice Commands

VII. CONCLUSION

The Virtual Drawing Board is an innovative application that delivers a smooth, touchless drawing experience by combining voice command processing with hand gesture recognition. [2], [3], [4]. This system provides an intuitive interface that enables users to draw, capture screenshots, clear the canvas, and create shapes without physically contacting a device. It achieves this by utilizing MediaPipe for hand tracking and SpeechRecognition for interpreting voice commands. This reimagined approach to digital drawing significantly improves human-computer interaction, making it more engaging, accessible, and intuitive [7], [14].

The application can be used by individuals with motor impairments because it was created with inclusivity as a priority. Furthermore, it fosters a touch-free, hygienic interaction model, which is particularly important in public or educational environments. The system showcases real-time responsiveness and precision in both voice and gesture recognition, encouraging creativity through fluid and natural controls. It illustrates how voice can enhance creative tools by facilitating easy shape creation via voice commands, providing a modern alternative to conventional input devices.

VIII. FUTURE SCOPE

Future development will concentrate on some major areas in order to enhance functionality and user experience:

Multi-Hand Gesture Support

It will support recognition of both hands at the same time so that complex gestures such as rotating and zooming can be achieved, while enabling multi-layered commands for professional users.

Advanced Shape Recognition

It will employ machine learning algorithms to read rough drawings as precise geometric figures, auto-correction of bad drawings in real-time.

Multi-User Collaboration

It will turn the system into a collaborative workspace where multiple users can draw on the same canvas simultaneously, thereby increasing its value in classroom settings and remote teams.

Cloud Integration

It will make seamless storage through tools such as Dropbox or Google Drive possible, enabling access from different devices and supporting file sharing and version control.

Voice Assistant Integration

It will include sophisticated voice technologies for natural interaction to make the app more accessible and user-friendly for everyone.

REFERENCES

- [1] Pranavi Srungavarapu, Eswar Pavan Maganti, Srilekha Sakhamuri, Sai Pavan Kalyan Veerada, Anuradha Chinta "Virtual Sketch using Open CV" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278- 3075 (Online), Volume-10 Issue-8, June 2021
- [2] Vantukal Reddy, Thumma Dhyanchand, Galla Vamsi Krishna, Satish Maheshwaram "Virtual Mouse Control Using Colored Fingertips and Hand Gesture Recognition", 11-12 Sep 2020 <https://ieeexplore.ieee.org/document/9242677>
- [3] "Air Canvas Application using OpenCV and NumPy in Python" International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: 08 | Aug 2021.
- [4] H.-Y. Chung, Y.-L. Chung, and W.-F. Tsai, "An efficient hand gesture recognition system based on deep cnn," in 2019 IEEE International Conference on Industrial Technology (ICIT). IEEE, 2019, pp. 853–858.
- [5] A. Rakowski and L. Wandzik, "Hand shape recognition using very deep convolutional neural networks," in Proceedings of the 2018 International Conference on Control and Computer Vision, 2018, pp. 8–12.
- [6] M. Verma, A. Gupta and S. K. Vipparthi, "One for All: An End-to-End Compact Solution for Hand Gesture Recognition," 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9534076.
- [7] Mod Ma'asum, F.F., Sulaiman, S. and Saparon, A., 2018. Real-time hand gesture recognition for embedded system. Journal of Electrical and Electronic Systems Research (JEESR), 12, pp.51-57.
- [8] Lech, Michal, Bozena Kostek, and Andrzej Czyzewski. "Virtual Whiteboard: A gesture-controlled pen-free tool emulating school whiteboard." Intelligent Decision Technologies 6.2 (2012): 161-169.
- [9] Sharma, S., Jakka, A. R., Srivastava, Y. V., Dhir, M. K., & Kumar, S. (2023). Hand Gesture Recognition with Efficiency Analysis. Available at SSRN 4490245.
- [10] Wang, Jindi, Ioannis Ivrisimtzis, Zhaoxing Li, and Lei Shi. "Hand gesture recognition for user-defined textual inputs and gestures." Universal Access in the Information Society (2024): 1-15.
- [11] Beño, L., Kučera, E., Drahoš, P., & Příbiš, R. (2024). Transforming industrial automation: voice recognition control via containerized PLC device. Scientific Reports, 14(1), 29387.

- [12] Hsieh, C. H., Lo, Y. S., Chen, J. Y., & Tang, S. K. (2021). Air-writing recognition based on deep convolutional neural networks. IEEE Access, 9, 142827-142836.
- [13] Saoji, S. U., Dua, N., Choudhary, A. K., & Phogat, B. (2021). Air canvas application using OpenCV and numpy in python. IRJET, 8(08), 2395.
- [14] Chandhan, T. H., Raj, N., Reddy, N. N. K., & AN, M. Z. (2023, August). Air canvas: hand tracking using OpenCV and MediaPipe. In 1st-International Conference on Recent Innovations in Computing, Science & Technology.
- [15] Li, S. A., Liu, Y. Y., Chen, Y. C., Feng, H. M., Shen, P. K., & Wu, Y. C. (2023). Voice interaction recognition design in real-life scenario mobile robot applications. Applied Sciences, 13(5), 3359.
- [16] Gayathri G Murali, Shukhaira Beegam T, Akhil Raju , A Amal, Tessy Abraham Azhikakathu, 2023, An AI – Based Virtual Drawing System Using Python, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 04.
- [17] Narasimha, V., Sandeep, K., Eshwar, V., & Zareen, S. (2024, March 19). *Whiteboard application using machine learning model*. International Journal for Research in Applied Science and Engineering Technology (IJRASET). ISSN 2321-9653.
- [18] F. Soroni, S. a. Sajid, M. N. H. Bhuiyan, J. Iqbal and M. M. Khan, "Hand Gesture Based Virtual Blackboard Using Webcam," 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2021, pp. 0134-0140.
- [19] Shirkar, R., Salunke, P., Singh, P., Shirke, S., & Gaydhane, J. (2023, April). Virtual mouse using hand gestures. International Journal of Scientific Development and Research (IJS DR), 8(4), Article IJS DR2304417.
- [20] Bag, K., Urankar, S., Yadav, A., & Chaudhari, R. (2023, April). Virtual whiteboard – A gesture-controlled pen-free tool. International Journal of Research Publication and Reviews, 4(4), 592–598. ISSN 2582-7421.