International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 12 | Dec 2024 www.irjet.net p-ISSN: 2395-0072

Sky Touch

Prof. Varsha Mashoria[1], Shravani Divekar[2], Satyam Singh[3], Priyanshu Kannaujiya[4], Shivam kumar[5], Aaryan Pandey[6]

[1]Professor of Dept . of AIML ,BE, LTCE, Maharashtra ,India
[2]Student of Dept .of AIML , BE , LTCE, Maharashtra ,India
[3]Student of Dept .of AIML , BE , LTCE, Maharashtra ,India
[4]Student of Dept .of AIML , BE , LTCE, Maharashtra ,India
[5]Student of Dept .of AIML , BE , LTCE, Maharashtra ,India
[6]Student of Dept .of AIML , BE , LTCE, Maharashtra ,India

Abstract - *Sky Touch is an open-source computer vision based project which aims to revolutionize the digital creativity. This cutting edge system allows the user to create art with a very simple and intuitive approach and with no use of traditional equipment such as a pen or a paper, or even digital tablets. Users can draw on a virtual canvas with their hands and use hand motions instead of techniques so that they do not have to create any physical contact with any of the tools necessary in modern art.*

The groundbreaking technology takes AI integration into art creation to a whole new level by allowing traditional drawing to be done with a few strokes on a digital interface. The main goal of the project is to achieve a similar feel of working with art however with the capabilities of a machine. The contactless tools enhance the experience of creating art without relying or having to engage with physical tools.

To properly meet the expectations of global users, Sky Touch needs to include advanced hand-recognition features into the program that support a vast amount of gestures. This would allow for natural interaction between the tool and the user leading to more convenient designing and artwork creating. This project not only can be used for professional artists or educators, it can also be aimed towards hobbyists which would enable wider range of creativity and interpretation.

Key Words **– Sky touch, AI, Hand -recognition, Artwork Virtual canvas .**

1.INTRODUCTION

In today's fast-evolving digital landscape, innovative applications that merge technology and art are gaining immense popularity. One remarkable example is Sky Touch, a platform that employs OpenCV to create an immersive and interactive digital painting experience. With this technology, users can draw, paint, or write in midair using hand gestures, simulating the sensation of working on a physical canvas or whiteboard.

At the heart of this project lies OpenCV, an open-source framework for computer vision, which enables real-time image analysis and processing. By combining OpenCV with gesture recognition techniques, Sky Touch tracks hand movements, determines the position and motion path of gestures, and translates them into vibrant digital creations on a virtual canvas.

This project represents a perfect blend of art, technology, and human-computer interaction. It redefines how we engage with digital art, offering a revolutionary, hands-free creative experience. Sky Touch caters to a diverse audience, including professional artists exploring novel tools, educators seeking engaging teaching aids, and enthusiasts who enjoy expressing creativity.

In this guide, we delve into the technical framework required to build a similar project, detailing the image processing, gesture recognition, and user interface design principles involved. Through step-by-step instructions and sample code, you'll learn how to bring your own digital canvas to life. Embark on this exciting journey to create a Sky Touch platform where your imagination becomes the brush, and the virtual canvas transforms into your creative playground.

1.1 Problem Statement

In the current system, there are no crayons or paints; instead, it works entirely with your fingers. The main challenge is identifying and distinguishing fingers from an RGB image without using a depth sensor. Other issues include handling movements beneath the drawing path and the absence of a fixed pen tip. Since the system traces every movement of the fingers, the resulting image can often appear abstract or unrefined.

A significant amount of coding effort is needed to allow the user to move the drawing location in real time with hand gestures. Additionally, users must learn and perform specific gestures correctly to operate the system smoothly. This project aims to find solutions to these challenges and improve the overall experience.

This concept also addresses broader issues. People with hearing impairments, for example, often face difficulties in communication, as sign language is not widely used. This system could help bridge that gap by allowing messages to be written in the air, with responses displayed digitally or through augmented reality.

Another problem it tackles is paper waste. A large portion of commercial and landfill waste comes from discarded paper used for writing and drawing. With on-air writing, users can draw or write efficiently without using physical materials, reducing paper consumption. This system provides a simple and eco-friendly way to communicate and express ideas, making it accessible and useful for a variety of purposes.

1.2 Purpose

The **Sky Touch** project aims to create an innovative and interactive digital drawing platform that allows users to paint and draw in the air using hand gestures and a camera system powered by OpenCV (Open Source Computer Vision Library). This project eliminates the need for traditional tools like a mouse or stylus, enabling users to create digital art through simple hand movements. By providing real-time visualization, users can see their artwork come to life instantly, making the experience more engaging and interactive.

The platform includes dynamic brush selection, allowing users to switch between various virtual brushes and colors for greater creative flexibility. It also features an intuitive and user-friendly interface, making it accessible to individuals of all ages and skill levels. OpenCV's gesture recognition technology ensures seamless hand movement tracking, translating gestures into precise drawing commands. Additionally, users can customize brush settings, line thickness, and other artistic elements, while also having the ability to save and export their digital creations.

Beyond creativity, **Sky Touch** serves as an educational tool, offering an engaging way to learn about computer vision and image processing concepts. The platform is designed to work across multiple operating systems, including Windows, macOS, and Linux, ensuring broad accessibility. As an open-source project, **Sky Touch** encourages community collaboration to enhance its features and capabilities continuously. This blend of technology and creativity makes **Sky Touch** an exciting and versatile tool for users of all backgrounds.

1.2 Scope

The scope of the **Sky Touch** project is broad and encompasses several key areas of development, usability, and impact. It combines technology and creativity, offering users a unique digital drawing experience through hand gestures, powered by OpenCV for real-time image processing and gesture recognition.

Creative Expression: The platform enables users to create digital art using their hands or a pointing device, eliminating the need for traditional input devices. This allows for a more intuitive and engaging way to interact with digital art, opening up creative possibilities for professional artists, hobbyists, and anyone interested in artistic expression.

Real-Time Visualization and Feedback: With real-time visualization of the drawing process, users can instantly see their creations come to life on the screen. This feature enhances the user experience, providing immediate feedback and allowing for a more interactive and dynamic creative process.

Customization and Flexibility: The project offers a wide range of customization options, from switching between virtual brushes and colors to adjusting brush settings, line thickness, and other artistic components. This flexibility allows users to create unique artworks, whether they are professionals or beginners.

Education and Learning: **Sky Touch** serves as an educational tool to introduce users to computer vision and image processing. It offers a fun and hands-on way to learn about these technologies while exploring their practical applications in art and creativity.

Cross-Platform Accessibility: By ensuring compatibility with multiple operating systems such as Windows, macOS, and Linux, **Sky Touch** expands its reach and makes it accessible to a wide range of users across different devices and platforms.

Community Collaboration and Open-Source Development: As an open-source project, **Sky Touch** encourages community involvement and collaboration. Users and developers can contribute to the platform's continuous improvement, adding new features, refining existing ones, and ensuring that the tool evolves with technological advancements.

Sustainability and Eco-Friendly Creativity: By allowing users to create art without physical materials like paper, **Sky Touch** promotes a sustainable and eco-friendly approach to digital creativity. It also reduces waste by

eliminating the need for traditional art supplies, providing an environmentally conscious alternative.

2 PURPOSED SYSTEM

The system architecture for the SKY TOUCH consists of several key components. The application enables users to create drawings or sketches in the air using their camera-captured movements. Below is a simplified description of its architecture:

User Interface (UI): This component serves as the front-end, allowing interaction with the user. The UI displays the drawing canvas for users to interact with hand offers controls to adjust brush size and color.

Camera Module: This module captures real-time video using the computer's camera. It provides a continuous video feed to the Image Processing Module, where the stream from the camera is processed. This module performs real-time image analysis to detect user gestures and movements, enabling them to draw in the air.

Fig 1- Working of sky touch

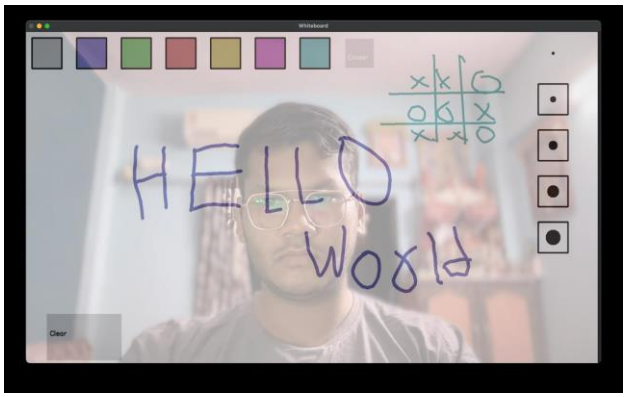

Fig 2- Working of sky touch

3. SYSTEM FLOW ARCHITECTURE

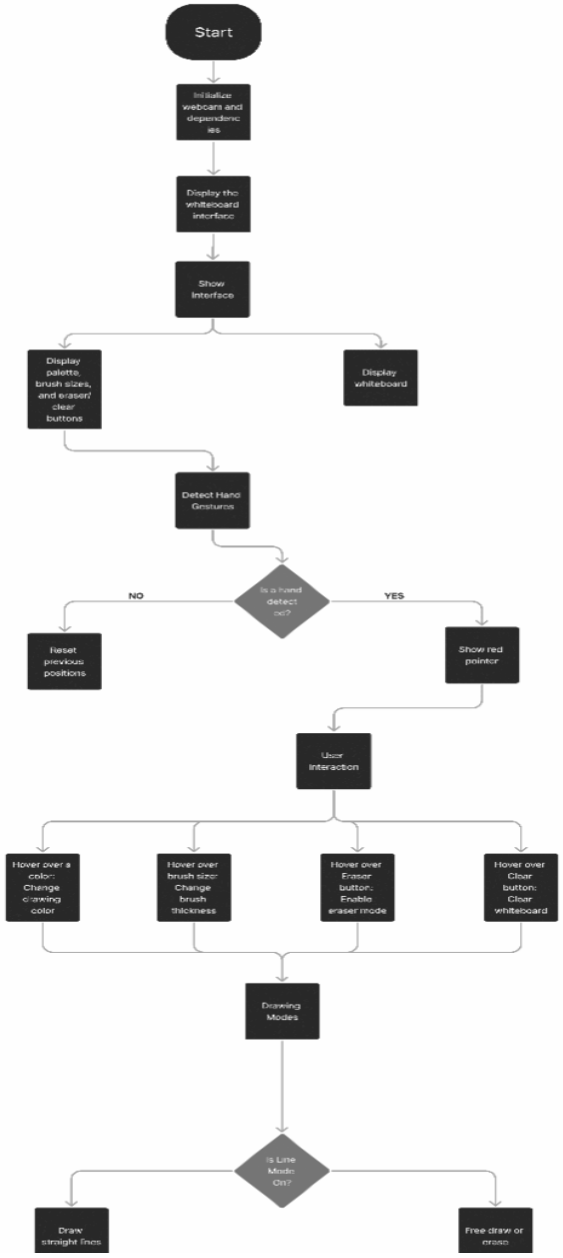

Fig 3. System flow architecture

4. SYSTEM REQUIREMENTS

Air Canvas is a digital whiteboarding and collaboration software. While general hardware and software requirements can be outlined, it's important to check the official documentation or system requirements from the developers, as they may be updated over time. Based on my last update in January 2022, the following are the basic hardware and software specifications for SKY Touch

4.1 Hardware Requirements:

Camera/Video Input

Webcam (HD or higher): A camera is crucial for real-time gesture recognition. The camera needs to capture the user's hand movements effectively.

Resolution: 720p or 1080p is ideal for accurate tracking and drawing.

Frame Rate: A higher frame rate (30 FPS or more) ensures smooth gesture tracking.

Processor (CPU)

Intel i5/i7 or equivalent (Recommended): A multi-core processor is ideal for handling real-time image processing and gesture recognition.

AMD Ryzen 5/7 (or equivalent) is also suitable.

Graphics Card (GPU) (optional but recommended)

NVIDIA GeForce GTX 1050 Ti or higher: While not strictly necessary, having a dedicated GPU can accelerate processing tasks, especially for real-time video and image processing.

Integrated GPU: Can be used if GPU isn't available, but performance may be slower.

Memory (RAM)

8GB RAM (minimum): 16GB or more is ideal for smoother performance when running multiple tasks or processing complex images.

Storage

Solid State Drive (SSD): Recommended for faster read/write speeds, especially when saving and exporting artwork files.

HDD: Suitable for lower budget systems, but performance may be slower.

4.2 Software Requirements

1)Operating Systems

Windows (preferred for development)

Linux

macOS

2)Programming Languages

Python (primary language for development)

3)Libraries and Frameworks

OpenCV: For computer vision and image processing (gesture recognition and real-time drawing).

MediaPipe: For hand tracking and gesture detection.

NumPy: For array manipulation and handling numerical data.

Tkinter or **PyQt**: For developing the graphical user interface (GUI).

Pillow: For image manipulation and saving artwork.

Py Auto GUI (optional): For controlling the mouse or simulating touch gestures.

TensorFlow (optional): For developing machine learning models for gesture recognition.

Matplotlib (optional): For visualizing data and charts.

PyInstaller or **cx_Freeze**: To package the Python application for distribution across different platforms.

Integrated Development Environment (IDE)

PyCharm or **Visual Studio Code**: Popular IDEs for Python development.

5. Literature Survey

Sno.	Name	Year	Working
1	EgocentricView Fingertip Detection for Air Writing Based on Convolutional Neural Networks by Yung-Han Chen, Chi-Hsuan Huang, Sin-Wun Syu, Tien-Ying Kuo, and Po-Chyi Su	2021	A practical air-writing scheme using a region-based convolutional neural network model for real-time fingertip localization. Simplified MobileNetV2 employed to reduce redundancy.
2	Air Doodle: A Realtime Virtual Drawing Tool by Soham Pardeshi, Madhuvanti Apar, Chaitanya Khot, Atharv Deshmukh	2022	Provides a real-time virtual drawing tool to replace notebooks for jotting down notes, offering ease of saving and sharing digital content.
3	Vision-Based Air-Writing Recognition System by Ramya P. and Rajasekaran M.	2020	Developed an air-writing system using deep learning for recognizing alphanumeric characters. Integrated CNN for feature extraction and classification.

4	Fingertip Detection for Wearable Devices by Nguyen Duc Thanh, Le Nguyen Ha, and Nguyen Thi Thu Thao	2021	Proposed a lightweight fingertip detection model for wearable devices using a custom-built neural network with efficient computational performance.
5	Deep Learning-Based Air-Writing Text Recognition by Shikha Bhardwaj and Sanjay Kumar	2023	Utilized LSTM and CNN for air-writing text recognition, focusing on accurate classification of sequential data in real time.
6	Hand Gesture Recognition for Air Writing using Leap Motion Sensor by Ananya Sharma and Rohan Singh	2019	Integrated Leap Motion sensor with a neural network for accurate detection and classification of air-writing gestures.
7	Air Handwriting Recognition Using Convolutional Recurrent Neural Networks by Michael Jones and Sarah Clark	2021	Combines CNN and RNN for recognizing air-written characters, enhancing accuracy by capturing spatial and temporal features of handwriting gestures.

6. PROJECT IMPLEMENTATION

6.1. Hand Detection and Tracking

Framework: MediaPipe Hands

Description: MediaPipe Hands detects and tracks hand landmarks in real time from the webcam feed.

Steps Involved:

The system processes video frames to locate the hand using a pre-trained machine learning model.

It identifies key points, such as the tips of fingers, and provides their coordinates.

These coordinates are used for gesture-based interaction with the whiteboard, such as drawing or selecting tools.

Output: Accurate hand landmark positions mapped to the canvas.

6.2 Gesture Recognition and Interaction Mapping

Purpose: Interprets hand movements and finger positions as commands for interacting with the whiteboard.

How It Works:

The system tracks the fingertip of the index finger to detect hovering or clicking over interactive regions like the color palette or brush selector.

It uses region-based mapping to determine which tool or setting the user interacts with.

Based on this interaction, the system adjusts settings like brush size or color.

6.3 Drawing with Smoothing

Technique: Line Drawing with Jitter Reduction

Purpose: Ensures smooth, continuous lines when the user draws on the canvas.

6.4 Color and Brush Selection

Approach: Region-Based Interaction

Objective: Allows the user to switch between different colors, brush sizes, and tools.

Implementation Steps:

Interactive areas are defined for each color and brush size.

When the user's fingertip enters one of these regions, the corresponding setting is activated.

The system provides visual feedback, such as highlighting the selected option.

Process:

The system records the fingertip's position across frames and connects them using lines.

6.5 Eraser Mode and Canvas Clearing

Techniques: Region Detection and Pixel Manipulation

Eraser Mode:

In this mode, the system draws white circles at the fingertip's location, erasing parts of the drawing.

Canvas Clearing:

If the user selects the "Clear All" option, the canvas resets to a blank white screen.

Result: These features make it easy to erase specific areas or start over.

6.6 Undo Functionality

Technique: Stack Data Structure

Purpose: Allows users to revert to previous canvas states.

How It Works:

A stack is used to store snapshots of the canvas after each drawing action.

When the user presses “Undo,” the system retrieves the most recent state from the stack and restores it.

6.7 Line Mode

Objective: Enables the user to draw straight lines between two points.

Process:

The user’s first fingertip position marks the starting point.

The system dynamically displays a preview of the line extending to the current fingertip position.

When the line is finalized, it becomes a permanent part of the canvas.

6.8. Blending the Canvas with the Webcam Feed

Technique: Alpha Blending

Purpose: Combines the camera feed and the whiteboard to create a semi-transparent overlay.

Implementation:

A blending formula is used to merge the two images: Blended Output=α×Whiteboard+(1−α)×Webcam Frame\text{Blended Output} = \alpha \times \text{Whiteboard} + (1 - \alpha) \times\text{WebcamFrame}Blended Output=α×Whiteboard+(1−α)×Webcam Frame

This provides the user with a live preview of their drawings alongside the webcam view.

7. FUTURE WORK

Future work of our project involves creating such an application, where programmers may encounter challenges with the precision of hand tracking, resilience of gesture identification, and performance enhancement. Further developments may include refining these aspects, improving gesture detection capabilities, and adding new functions such as sharing and storing images. Additional features, such as an eraser and tools for drawing shapes, are planned for the "SkyTouch" application.

In the future, the application aims to include enhanced shape-drawing capabilities and evolve into a full-fledged web application or mobile app. In conclusion, developing an application called "SkyTouch" with OpenCV presents an exciting opportunity to merge technology and art. By enabling users to express themselves through digital creativity, the program can be both entertaining and educational. Applications like this have a promising future thanks to the continual innovation and progress in computer vis**ion,** despite potential challenges.

8.CONCLUSION

In conclusion, *SkyTouch* offers a groundbreaking way to blend technology with creativity, providing users with a unique platform for gesture-based digital art. Built on OpenCV, the application opens up new possibilities for artistic expression while addressing challenges like improving hand-tracking accuracy and gesture recognition. The planned enhancements, including shape-drawing tools and expansion into a full-scale web or mobile application, highlight its potential for growth and versatility. With continuous advancements in computer vision, *SkyTouch* is poised to become a valuable tool for fostering innovation and creativity in the digital space.

8. REFERENCES

[1] Y. Huang, X. Liu, X. Zhang, and L. Jin, "A Pointing Gesture Based Egocentric Interaction System: Dataset, Approach, and Application," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, pp. 370-377, 2016.

[2] P. Ramasamy, G. Prabhu, and R. Srinivasan, "An economical air writing system converting finger movements to text using a web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, pp. 1-6, 2016.

[3] C.-H. Hsieh, Y.-S. Lo, J.-Y. Chen, and S.-K. Tang, "Air-Writing Recognition Based on Deep Convolutional Neural Networks," IEEE Access, vol. 9, pp. 142827-142836, 2021.

RESEARCHGATE

[4] S. Mukherjee, A. Ahmed, D. P. Dogra, S. Kar, and P. P. Roy, "Fingertip Detection and Tracking for Recognition of Air-Writing in Videos," arXiv preprint arXiv:1809.03016, 2018.

ARXIV

[5] V. Jain and R. Hebbalaguppe, "AirPen: A Touchless Fingertip Based Gestural Interface for Smartphones and Head-Mounted Devices," arXiv preprint arXiv:1904.06122, 2019.

ARXIV

[6] P. Roy, S. Ghosh, and U. Pal, "A CNN Based Framework for Unistroke Numeral Recognition in Air-Writing," arXiv preprint arXiv:2303.07989, 2023.

ARXIV

[7] A. Tripathi, A. K. Mondal, L. Kumar, and P. A. P., "ImAiR: Airwriting Recognition framework using Image Representation of IMU Signals," arXiv preprint arXiv:2205.01904, 2022.

ARXIV

[8] S. Sonpda and Y. Muraoka, "Air-Writing Recognition Using Wearable Video Camera," Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, pp. 2105-2110, 2015.

[9]M. Setiawan and R. Pulungan, "Air-Writing Recognition Using Leap Motion and Deep Belief Networks," 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Bali, pp. 1-6, 2017.

[10] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," Artificial Intelligence Review, vol. 43, no. 1, pp. 1-54, 2015.

These references provide a comprehensive overview of various approaches and technologies used in air-writing and gesture recognition systems.