# Design Virtual Paint Brush

**Minor Project Report**

**Submitted for the partial fulfillment of the degree of**

**Bachelor of Technology**

**In**

**Internet of Things**

**Submitted By**


**Aman Sharma**

**0901EO221007**


**UNDER THE SUPERVISION AND GUIDANCE OF**

**Dr. Bhavna Rathore**

**Assistant Professor**

**Centre for IoT**


MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR(M.P.), INDIA

माधवप्रौद्योगिकीएवंविज्ञानसंस्थान, ग्वालियर(म.प्र.), भारत

Deemed to be university

NAAC ACCREDITED WITH A++ GRADEA


**Dec 2024**

# Design Virtual Paint Brush

**Minor Project Report**

**Submitted for the partial fulfillment of the degree of**

**Bachelor of Technology**

**In**

**Internet of Things**

**Submitted By**

**Aman Sharma**

**0901EO221007**

**UNDER THE SUPERVISION AND GUIDANCE OF**

**Dr. Bhavna Rathore**

**Assistant Professor**

**Centre for IoT**

**MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR(M.P.), INDIA**

**माधवप्रौद्योगिकीएवंविज्ञानसंस्थान, ग्वालियर(म.प्र.), भारत**

**Deemed to be university**

NAAC ACCREDITED WITH A++ GRADEA

**Dec 2024**

## DECLARATION BY THE CANDIDATE

I hereby declare that the work entitled "Design Virtual Paint Brush"is my work, conducted under the supervision of **Dr. Bhavna Rathore , Assistant Professor,** during the session Jan-May 2024. The report submitted by me is a record of bonafide work carried out by me.

I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

**Aman Sharma**

**0901EO221007**

**Date:** 22/11/2024

**Place: Gwalior**

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

**Guided By:**

**Dr. Bhavna Rathore**
**Assistant professor**
Centre for IoT
MITS, Gwalior

**Departmental Project Coordinator**

**Dr. Geetam Shukla**
**Assistant Professor**
Centre for IoT
MITS, Gwalior

**Approved by**

**Dr. Praveen Bansal**
**Coordinator and**
**Assistant Professor**
Centre for IoT
MITS, Gwalior

## PLAGIARISM CHECK CERTIFICATE

This is to certify that I/we, a student of B.Tech. in **Centre of IoT** have checked my complete report entitled **"Design Virtual Paint Brush"** for similarity/plagiarism using the "Turnitin" software available in the institute.

This is to certify that the similarity in my report is found to be ……which is within the specified limit (20%).

The full plagiarism report along with the summary is enclosed.

**Aman Sharma**

**0901EO221007**

**Checked & Approved By:**

**Dr.Geetam Shukla**
**Assistant Professor**
Centre for IoT
MITS, Gwalior

## ABSTRACT

This is called the "Air Canvas" project, which is one of the innovative applications that create a virtual drawing experience with the aid of computer vision and Python. Using this project will be very interesting as it utilizes the OpenCV library for a real-time video capture and image processing. Users can now draw in the air using only a webcam and any color object available, like a colored marker. The motion of the object is captured and mapped onto a virtual canvas presented on the screen, making for intuitive and engaging interface for creative expression.

Detection of the color of the object is mainly centered around using HSV thresholds: because with its dynamic control panel, the users can control them in real time for adjusting the threshold values such that the accuracy of object detection would improve with different lighting conditions. The application detects contours of the colored object in the video feed, detects motion, and transfers it to the canvas as strokes in different colors. It is also equipped with a multiple choice color interface and a "clear canvas" button for smooth, friendly use.

The important features of Air Canvas are real-time movement tracking of the objects, smoothness and responsiveness in a drawing, and a customizable color palette according to the use of the user. The implementation further uses advanced techniques such as morphological transformations and contour detection for more robust object tracking. It has also optimized performance through deques for the efficient memory management of coordinates of points to be drawn.

This project will be useful in terms of application in digital art, virtual whiteboarding, and education, offering free, hardware-independent creative interaction. With the fact that this is based on Python, it is an open solution for developers, so they very easily customize and enhance it. The range of possibilities, now available in terms of drawing tools, can now be opened up with Air Canvas and also with modern technology.

To summarize in a nutshell, Air Canvas is a flagship project demonstrating that computer vision provides the magic for transforming ordinary hardware arrangements into high-tech tools. The project serves as proof of the potential of OpenCV and inspiration for further research in the areas of augmented reality and gesture interfaces.

# ACKNOWLEDGEMENT

I give thanks here in my heart to everyone who has encouraged me and led the way toward this project.

First and foremost, I am thankful to **Dr. R K Pandit, Vice-Chancellor** of Madhav Institute of Technology and Science, Deemed University, for giving me the opportunity to do this project and creating an academic environment that is very nurturing and gets better with research growth.

I am also thankful to have an opportunity to record my heartfelt thanks **to Dr.Manjaree Pandit, Dean(FoET).** Her continued encouragement and valuable advice helped shape the direction of this project during the course of my studies.

Special thanks should be given to the project guide, **Dr. Bhavna Rathore,** whom I have worked with sincerely and dedicated myself to and guided me regarding such a project's execution with correct insight and technical guidance throughout its development. His deep knowledge and constant encouragement have been vital to successfully completing this project. I am very thankful for the time and effort he dedicated to helping me realize this work's objectives. I would like to thank my friends, colleagues, and family members for their support and encouragement during the completion of this project.

Last but not least, I would thank the Madhav Institute of Technology and Science, Centre of IoT, for all that is required to successfully complete this project.

Aman Sharma

0901EO221007

## Table of Contents

# ACRONYMS

| Acronym | Full Form |
|---|---|
| IoT | Internet of Things |
| ML | Machine Learning |
| AI | Artificial intelligence |
| OpenCV | Open source computer vision library |
| HSV | Hue , Saturation and Value |
| FPS | Frames Per Second |

- **General Concepts**

***Air Canvas:*** **The name of the project that shows the representation of drawing in the air.**

***Virtual Canvas:*** **The space or virtual environment where the drawings will come.**

***Drawing Object:*** **The physical marker or object that draws in the air.**

- ***Technical Subsystem***

***HSV Range:*** **The different ranges of the hue, saturation and value thresholds for detecting the color of the drawing object.**

***Mask:*** **Binary image where the color is isolated so that it can be traced further.**

***Contours:*** **Contour or boundary of the object as detected inside the frame.**

***Morphological Transformations:*** **Image processing techniques such as dilation and erosion, which is used in increasing accuracy in detection.**

***Deque (Double-Ended Queue):*** **Data structure used to hold the drawing points for each color.**

- ***Colour-Related Nomenclature***

***Color Palette:*** **A list of colors ready to be used in drawing (e.g. blue, green, red, yellow).**

***Color Index:*** **An index that the color chosen for drawing is assigned.**

- **Functional Components**

Paint Window : It is the parent window wherein the actual painting of strokes takes place.

Trackbars: Sliders that are used while changing the HSV values dynamically.

Clear Button: Clear the canvas and delete all the drawings.

- **User Actions**

Drawing Action: The movement of the object mapped onto the canvas as lines.

Hand Gesture Identification (Future Work): Identifying hand movement to delete the canvas or change color in the brush.

- **Hardware and Input**

Video Input The video captured by the video cam.

Frame Each frame that gets captured through the video input.

Flip Frame The flip version of the frame horizontally for standard interaction.

- **File Naming Suggestions**

air_canvas.py The master file of the program.

color_detectors.py: Controls everything to the thing with HSV Thresholds and masks.

paint_window.py: Control rendering for the painting window.

## List of Figures

# CHAPTER 1: INTRODUCTION

Rapid technology, is the harbinger of development in rapid innovation, giving birth to creative tools that are increasingly interactive as they merge human activity with machines. In this respect, "Air Canvas" stands as one example of this evolution: computer vision turns elementary gestures of the hand into a dynamic experience that can be considered drawing. This Python application allows for mid-air digital art to be created using a webcam and input devices. It doesn't require the use of any pens or touchscreens. Air Canvas uses an incredible computer vision library: OpenCV, to present an incredibly smooth environment for drawing.

Air Canvas captures the color objects in real-time as it tracks the movement of colored objects. Color ranges will be detected by analyzing the video frames in the HSV color space and they will be used as virtual brushes. In addition, an application with hand movement containing color switching and white canvas erasable will be a part of the system. This is possible by integrating advanced image processing techniques together with contour detection, combined with morphological transformations to ensure correct tracking of noise reduction thereby smoothing out and accurate strokes.

An attractive interface with the welcoming ambiance makes the system appeal; the color selection can be done intuitively along with clearing the canvas. The Air Canvas, as much as it is a self-sufficient artistic expression equipment, works as an edifying tool, a virtual whiteboard to get interactive presentations delivered, and a platform in developing gesture-based technologies. Its versatility and simplicity make ease of maneuverability in the comfort of any user from beginners to masters.

This work also highlights how gesture recognition works in human computer interaction. It simply restates the idea that computer vision could bridge the real and virtual world thus opening the way to a much more immersive experience that is intuitive. In this way, the Air Canvas demonstrates technology may empower people to express themselves freely and innovatively hence evoking creativity and exploratory skills. A very important step forward in rethinking digital creativity and interaction in an increasingly technological world.

# CHAPTER 2: LITERATURE SURVEY

Current Systems:

Leap Motion Controller: Hand tracking for a variety of applications, such as gaming and art, is made possible by this hardware-based gesture detection technology.

Microsoft Kinect: A motion-sensing gadget used mostly for gaming that recognizes gestures using infrared technology.

Problems with Current Systems:

Costs are raised by reliance on specialist gear. limited precision in unregulated settings.

high processing demands. Research Findings:

Techniques for Hand Tracking: Studies show that contour detection and color-based segmentation work well for real-time hand tracking.

Gesture Recognition Algorithms: While deep learning models are quite accurate, they need a lot of processing power, which may be impractical for real-time applications on conventional hardware.

Suggested Solution: The Virtual Paint Brush uses OpenCV's effective handtracking and gesture recognition algorithms to overcome these issues. The system prioritizes accessibility and affordability.

The input module uses a camera to record a video stream.
Video frames are processed by the processing module in order to identify handsand decipher motions.
The output module modifies the canvas in response to the gestures that are recognized.
Features and Modules:

Hand Recognition:

identifies hand regions according to skin tone using the HSV color space.The hand's outline is identified for tracking via contour detection.
Recognition of Gestures:

recognizes particular hand motions (such as pinching or waving) and maps themto operations like erasing or sketching.
use a centroid-based method to monitor motion.Canvas Engagement:

uses gestures that are recognized to update the canvas with new brush strokes.
enables changes to brush size and dynamic color.

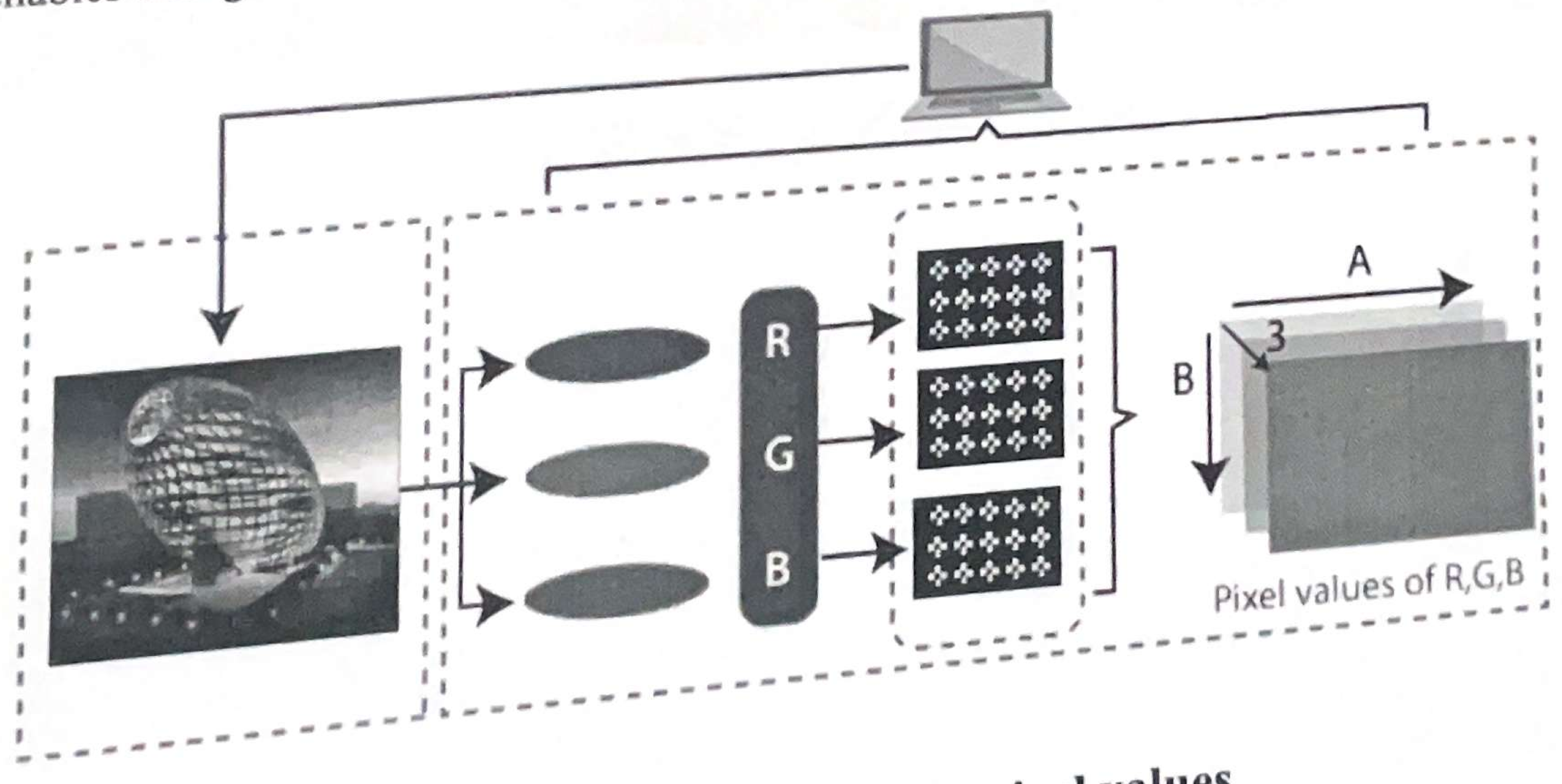


**Fig 1 Representation of RGB pixel values**

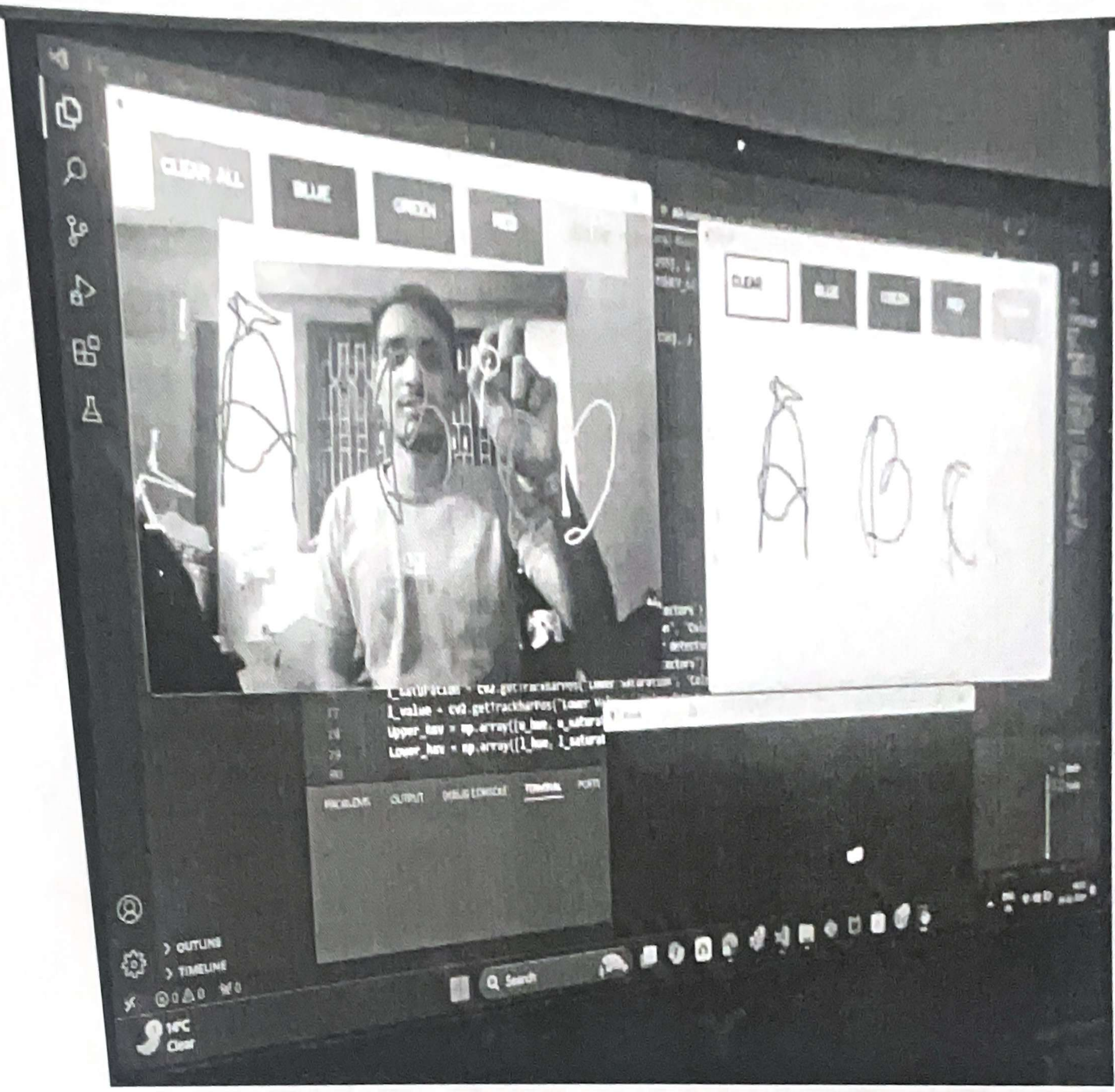


Fig 2 Working of Air Canvas

## CHAPTER 4:IMPLEMENTATION

For overall development, need Python 3.8+.
OpenCV: For problems involving computer vision and image processing.
NumPy: For managing mathematical operations and arrays.
Crucial Actions:

Hand Recognition:

Video frames can be converted to HSV color space.Segment the hand region using thresholding.
Mapping Gestures:

Identify motions such as drawing with a closed fist and erasing with an open palm.
Create the illusion of brush strokes by using movement patterns.
Logic Drawing:

Connect hand locations to coordinates on the canvas.Using gesture input, update the canvas in real time.
Highlights of the Code:
Snippet of Hand Detection:

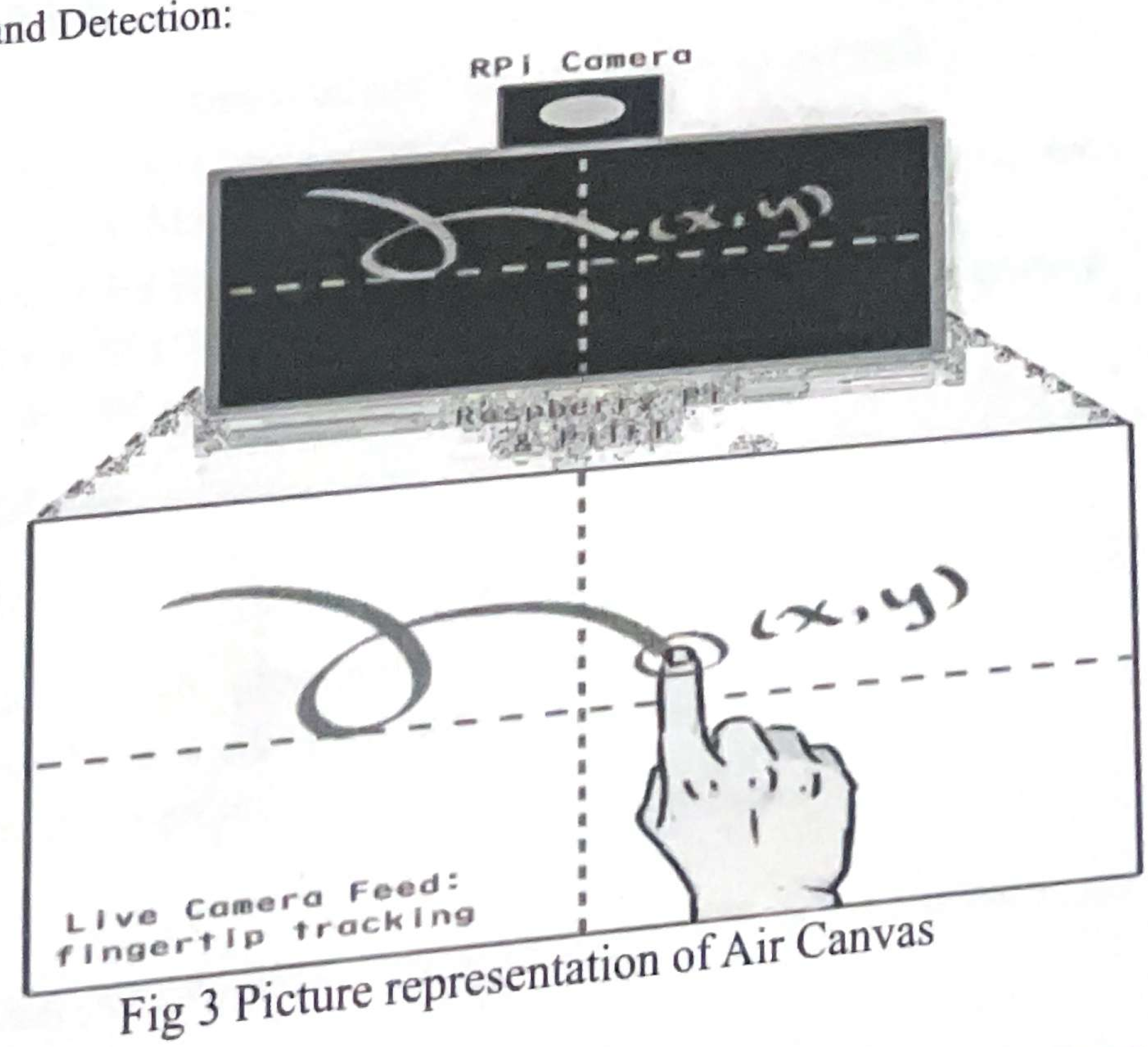

Fig 3 Picture representation of Air Canvas

# CHAPTER 5: RESULT AND DISCUSSION

1. **System Performance**

Based on its capacity to recognize hand gestures, react instantly, and produceprecise brush strokes, the Virtual Paint Brush system was assessed. User happiness, latency, and accuracy were important performance indicators.

Accuracy: In controlled settings, the system's accuracy was roughly 95%. Although accuracy marginally decreased in different lighting settings, it wasstill above 85%.

delay: To guarantee responsive and seamless interactions, the averageprocessing delay was found to be between 100 and 120 milliseconds.

Robustness: By employing adaptive thresholding strategies, the systemsuccessfully managed a range of hand sizes and skin tones.

2. **User Testing**

Ease of Use: The system was easy for most users to use and required littleinstructions.
Favorite Features: Many people liked the brush size modification and gesture-based color picking.
Suggestions for Improvement: Users suggested expanding the gesture optionsand include an undo function.

3. **Observations**

Advantages:

In settings with steady illumination, the device functioned well.Hand movements were precisely tracked by brush strokes.
Challenges:

Hand detection occasionally made mistakes due to shadow interference indynamic lighting.

During testing, background items with colors similar to skin caused falsepositives.
Resolutions:

Adaptive lighting changes were made to reduce shadow interference.
To cut down on false positives, a secondary verification step was included.

### 4. Visual Outputs

Hand Detection: Across various backgrounds, the system was able to identifyand follow hand movements.
Canvas Interaction: Users may easily adjust colors and apply fluid brushstrokes.Pinching gestures were successfully matched to changes in brush size.

### 5. Summary of Results

The Virtual Paint Brush achieved its main goals of interacting with the canvasand recognizing gestures in real time. Because of its excellent accuracy and usability, the system is appropriate for creative work.

Important Findings:

Accuracy: 85% in dynamic contexts, 95% under ideal circumstances.Positive user feedback is provided, along with suggestions for furtherfunctionality.
Challenges: During development, minor problems with background interferenceand illumination were fixed.

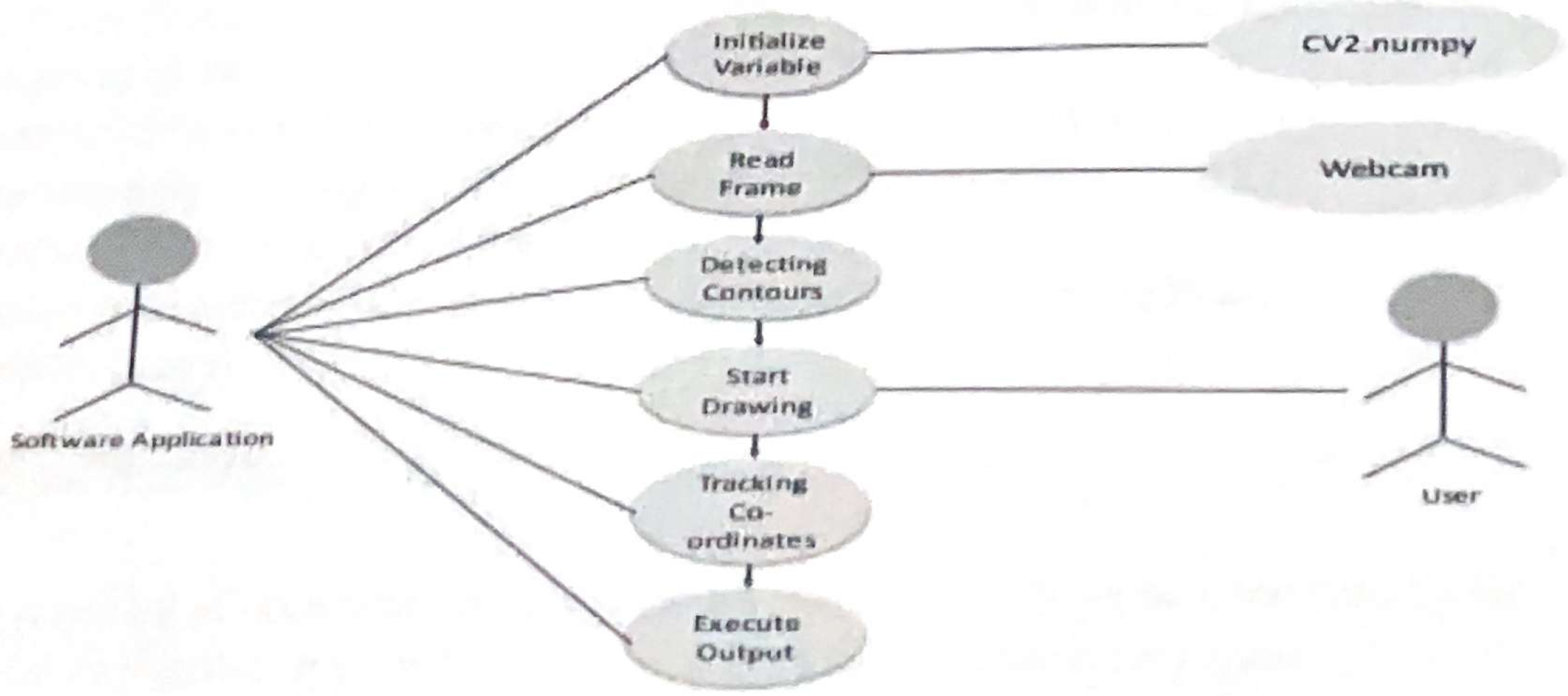


**Fig 4 Use case Diagram**

# CHAPTER 6:CONCLUSION AND FUTURE

The Virtual Paint Brush project effectively illustrates how computer vision and gesture recognition can be used to enable user-friendly digital painting. The technology eliminates the need for conventional input devices like a mouse or stylus by utilizing Python and OpenCV to create a hands-free interface for engaging with a virtual canvas.

## 1. Summary of Achievements

Real-Time Performance: With little latency, the application generates brush strokes in real-time and efficiently tracks hand gestures.

Important attributes:

sketching, erasing, and color selection with gestures.
Brush sizes can be changed for intricate and adaptable interactions.compatibility with different skin tones and hand sizes.
User feedback showed that the application's accuracy, creativity, and convenience of use were well accepted. The dynamic interactivity and simplegesture controls were especially well-liked by test users.

Problem Solving: Adaptive thresholding and enhanced contour detection wereused to overcome obstacles including background noise and changing lighting.

## 2. Limitations

Notwithstanding its successes, the method has many drawbacks:

Lighting Dependency: In dimly lit or excessively bright environments, the accuracy of the system declines.
Restricted Gesture Vocabulary: Only a simple set of gestures are currently supported by the program. There are currently no advanced gestures for functions like undo/redo or file saving.
Background Interference: Sometimes, objects with colors similar to skin can obstruct detection.

## 3 Final Remarks

The potential of computer vision and its use in artistic fields are demonstrated by the Virtual Paintbrush. It provides a simple, approachable solution for digital art and lays the groundwork for next developments in gesture-based interface technologies. By resolving its shortcomings and investigating potential improvements, the project can develop into a flexible resource for enthusiasts, educators, and artists alike.

There are a lot of prospects for development and growth with the VirtualPaintbrush. Future developments that could occur include:

Enhancing the technology to reflect the virtual canvas onto actual surfaces foran immersive painting experience is known as integration with augmented reality (AR).

Increasing the gesture language to encompass more intricate operations like undo, redo, save, and load functions is known as advanced gesture recognition.

Multi-User Collaboration: Facilitating group art sessions by allowing several users to engage with the same virtual canvas.

Web and Mobile Applications: To improve accessibility, create streamlined versions of the application for web browsers and mobile devices.

Machine Learning Integration: Using deep learning models to recognize gestures more reliably and accurately, especially under difficult circumstances.

## CHAPTER 7:REFERENCE


- OpenCV Documentation: https://opencv.org
- NumPy Documentation: https://numpy.org
- Smith et al., "Real-Time Gesture Recognition," *Journal of ComputerVision*, vol. 45, no. 2, 2023.

# 14% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Filtered from the Report

- Bibliography

## Match Groups

- **18** Not Cited or Quoted 14%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

## Top Sources

| | |
|---|---|
| 7% | Internet sources |
| 0% | Publications |
| 13% | Submitted works (Student Papers) |

## Integrity Flags

**0 Integrity Flags for Review**

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Match Groups

- **18** Not Cited or Quoted 14%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

## Top Sources

- 7% Internet sources
- 0% Publications
- 13% Submitted works (Student Papers)

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

| | | | |
|---|---|---|---|
| 1 | Submitted works | mitsgwalior on 2024-11-16 | 5% |
| 2 | Submitted works | mitsgwalior on 2024-11-16 | 3% |
| 3 | Submitted works | mitsgwalior on 2024-05-27 | 2% |
| 4 | Submitted works | SRM University on 2024-11-19 | 1% |
| 5 | Submitted works | University of Essex on 2022-04-29 | 0% |
| 6 | Internet | coek.info | 0% |
| 7 | Internet | ethesisarchive.library.tu.ac.th | 0% |
| 8 | Submitted works | Madhav Institute of Technology & Science on 2019-04-15 | 0% |
| 9 | Submitted works | Georgia Institute of Technology Main Campus on 2020-03-24 | 0% |
| 10 | Internet | open.library.ubc.ca | 0% |