PAPER NAME	AUTHOR
irtual paint pdf (1).pdf	S G

WORD COUNT	CHARACTER COUNT
3142 Words	20223 Characters
PAGE COUNT	FILE SIZE
27 Pages	785.5KB
SUBMISSION DATE	REPORT DATE
Nov 19, 2024 2:37 PM GMT+5:30	Nov 19, 2024 2:37 PM GMT+5:30

15% Overall Similarity

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

- 9% Internet database
- Crossref database
- 14% Submitted Works database
- 0% Publications database
- Crossref Posted Content database

Excluded from Similarity Report

- Bibliographic material
- Small Matches (Less then 10 words)

ANNEXURE

SELF-EVALUATION OF INTERNSHIP/ PROJECT

	Start date – End date (DD/MM/YY) - (DD/MM/YY)	Progress of Internship/ Project
-1	05/08/24 - 31/08/24	Topic Selection and Project Planning
-2	01/09/24 - 30/09/24	Hardware selection and Prototype Development
-3	01/10/24 31/10/24	Data Analytics and Results
-4	01/11/24 – 18/11/24	Documentation of the project

Dr. Soumyajit Ghosh
Assistant Professor

TURNITIN PLAGIARISM REPORT

- Insert a Scanned Copy of the Front pages duly signed by the Candidate, Supervisor, Departmental Turnitin Coordinator, and HoD with Seal

References

...enCV Documentation: https://opencv.org

...ediaPy Documentation: https://mediapy.org

... Smith et al., "Real-Time Gesture Recognition," *Journal of Computer ...tion*, vol. 45, no. 2, 2023.

CHAPTER 7: FUTURE SCOPE

lot of prospects for development and growth with the Virtual
Future developments that could occur include:

the technology to reflect the virtual canvas onto actual surfaces for
ove painting experience is known as integration with augmented
(R).

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

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

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

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

CHAPTER 6: CONCLUSION

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.

6.1 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 simple gesture controls were especially well-liked by test users.

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

6.2 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.

6.4 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 digi

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

5.4 Visual Outputs

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

5.5 Summary of Results

The Virtual Paint Brush achieved its main goals of interacting with the canvas and 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 further functionality.
Challenges: During development, minor problems with background interference and illumination were fixed.

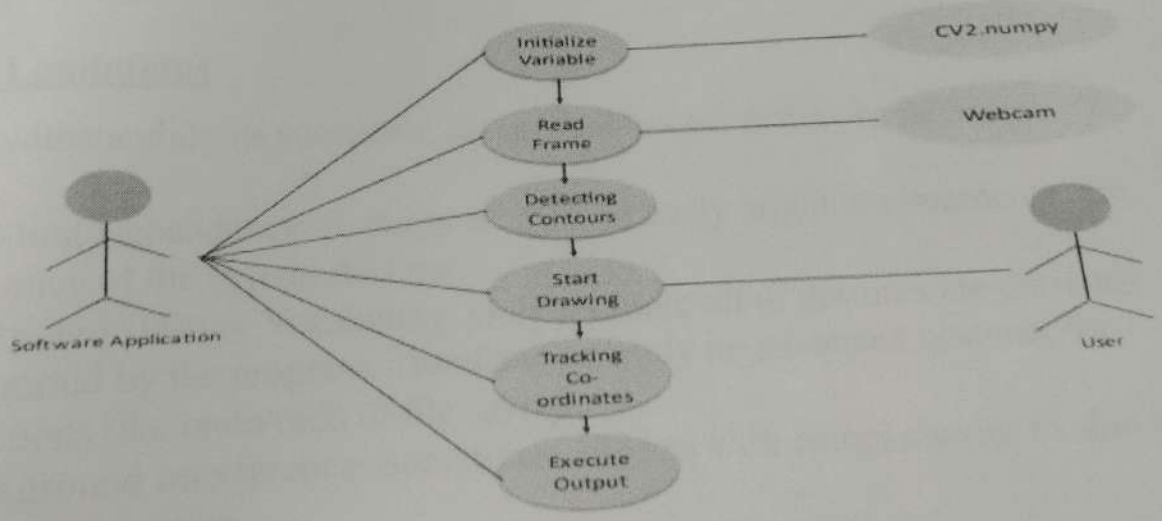

Fig 4 Use case Diagram

5.1 System Performance

Based on its capacity to recognize hand gestures, react instantly, and produce precise 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 was still above 85%.

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

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

5.2 User Testing

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

5.3 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 in dynamic lighting.
During testing, background items with colors similar to skin caused false positives.
Resolutions:

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:

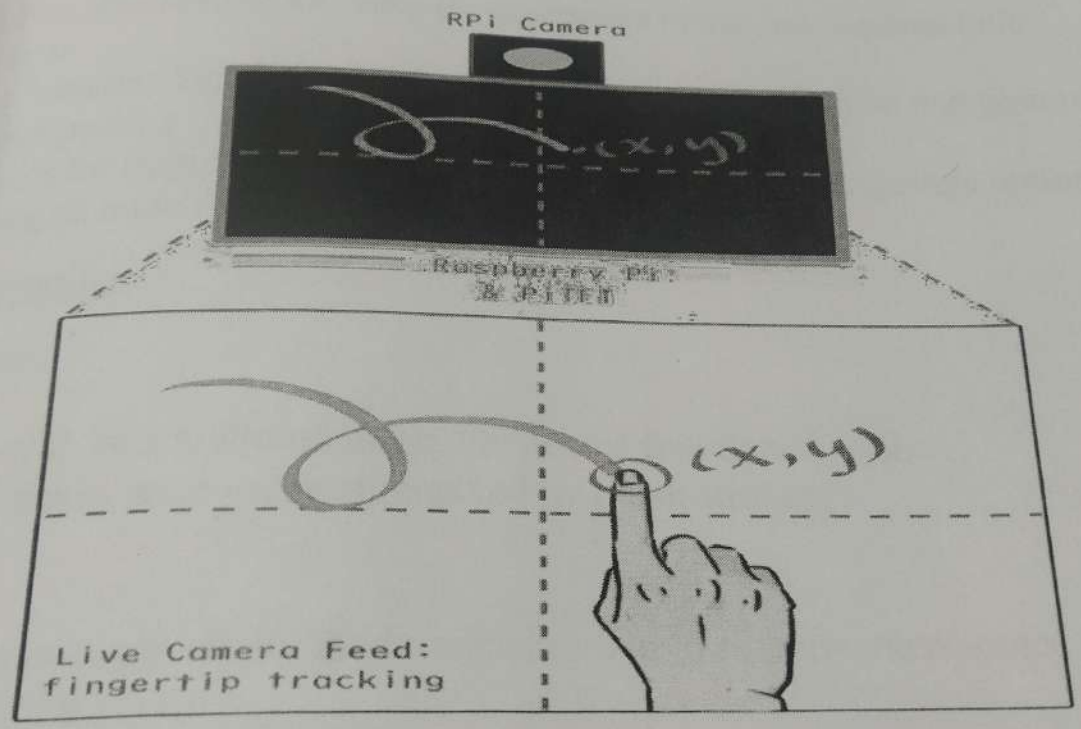

Fig 3 Picture representation of Air Canvas

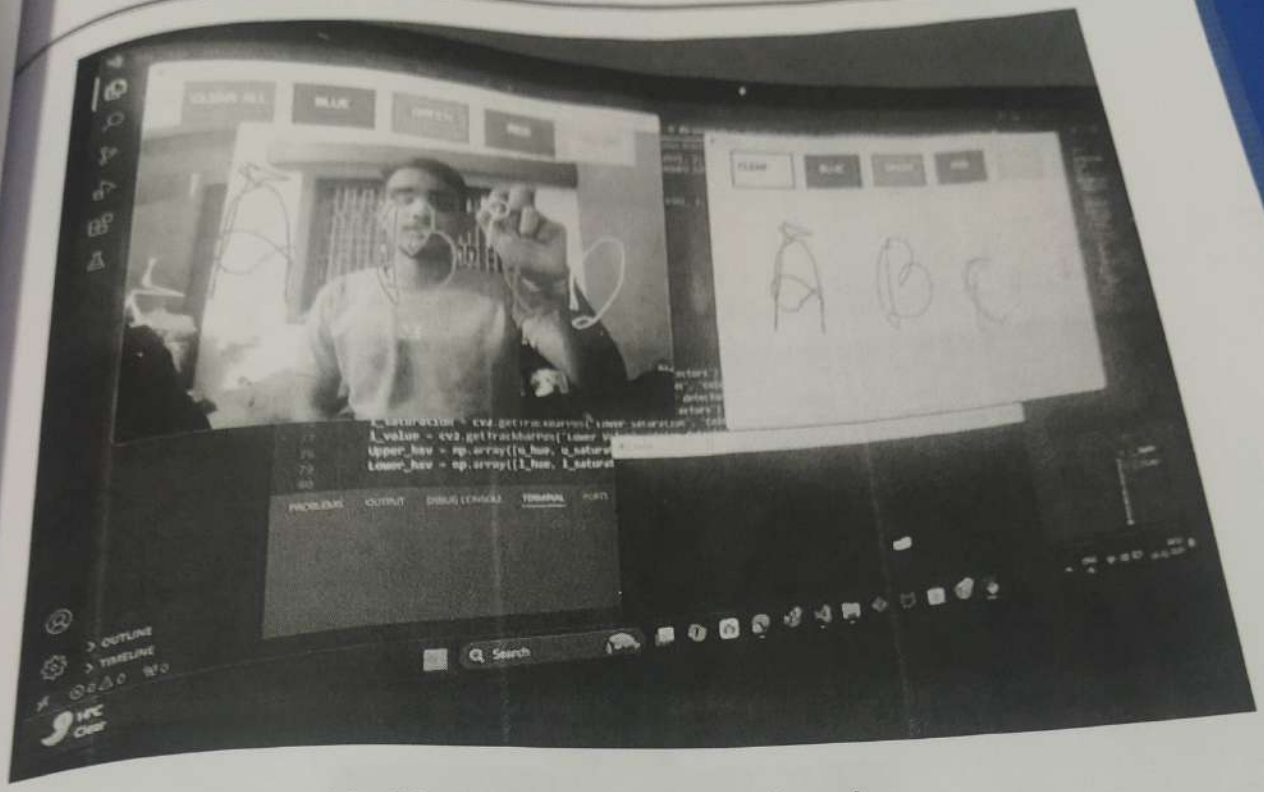

Fig 2 Working of Virtual Paint Brush

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 hand tracking and gesture recognition algorithms to overcome these issues. The system prioritizes accessibility and affordability.

- **Chapter 4: Implementation** explains the development process, including algorithms and code structure.
- **Chapter 5: Results and Analysis** evaluates the system's performance and user feedback.
- **Chapter 6: Conclusion** summarizes the project's achievements and explores potential future enhancements.

1.4 Expected Outcomes

A fully working application that can convert hand motions into brush strokes on a digital canvas while tracking them in real-time.
high user satisfaction as a result of the system's ease of use and interaction.
insights into real-time gesture recognition's problems and potential solutions.

1.5 Technical Relevance

This project demonstrates how computer vision algorithms are used in practice, particularly those that involve:

Hand Detection: Methods like contour detection and color-based segmentation.
Mapping hand gestures to drawing motions is known as gesture recognition.
Effectively updating and showing brush strokes on a digital canvas is known as canvas rendering.
The project shows how open-source tools can be used to create creative apps by utilizing Python and OpenCV. Additionally, it draws attention to how gesture-based systems might improve accessibility and engagement in a variety of contexts.

1.6 Significance

The significance of the Virtual Paintbrush project stems from:

It removes the requirement for specialist hardware, democratizing access to digital art tools.
It provides a platform for learning about computer vision and gesture detection.
It supports hands-free, user-friendly interfaces, which is in line with global trends in human-computer interaction.

1.7 Structure of the Report

- **Chapter 2: Literature Survey** provides an overview of existing systems, techniques, and research in gesture recognition and digital art tools.
- **Chapter 3: System Design** discusses the architecture and components of the Virtual Paint Brush application.

1.2 Motivation

The Virtual Paint Brush was created with the intention of making digital creativity more approachable and accessible to everyone. Physical input devices present obstacles that gesture-based controls remove, giving users—including those with physical disabilities—new options.

Both enthusiasts and creative professionals frequently look for tools that combine sophisticated capabilities with ease of use. This need is met by the Virtual Paint Brush, which offers a user-friendly, interactive digital painting environment. The study also demonstrates the useful applications of continuing developments in machine learning and computer vision.

1.3 Objectives

The following are the project's main goals:

1. Create a digital painting tool that uses gestures: Make an application that enables users to utilize hand gestures to sketch on a virtual canvas.
2. Put essential characteristics into practice:
gesture recognition for color selection, erasing, and drawing.
3. Brush sizes can be changed for more flexibility and control.
4. Assure performance in real time: Make the system as accurate and low latency as possible.
5. Provide a solution that is easily accessible: To make the system affordable and accessible, use open-source libraries and common webcams.

The **Virtual Paint Brush** project is inspired by the advancements in computer vision and human-computer interaction. Traditional drawing tools are effective but require physical input devices, which may not always be accessible or intuitive. This project aims to transform digital art by enabling users to interact directly with the canvas using hand gestures.

This application employs Python as the programming language and OpenCV as the core library for image processing. The system captures real-time video streams, processes the input to detect hand gestures, and translates those gestures into painting actions. By integrating features such as color selection, brush size adjustment, and gesture controls, the **Virtual Paint Brush** offers a comprehensive digital painting experience.

1.1 Background

Human-computer interaction has evolved beyond conventional input devices like keyboards and mouse as a result of technological advancements. Gesture-based systems, which enable users to interact with gadgets via hand movements, have emerged in response to the need for natural, intuitive interfaces. In industries like gaming, virtual reality, healthcare, and the creative arts, this method is especially helpful.

Computer peripherals like styluses, touchscreens, and mice are necessary for digital painting programs like Adobe Photoshop and Microsoft Paint. Even though these tools work well, not everyone can utilize them, and users who prefer more direct, organic interactions may find them difficult to use. By providing a hands-free solution that allows users to create art with basic hand gestures, the Virtual Paint Brush aims to close this gap.

List of Figures

Technical Abbreviations

HSV, or hue, saturation, and value color space, is a color-based segmentation technique used in computer vision.

Frames Per Second, or FPS, is a crucial metric for real-time applications since it measures how many video frames are handled in a second.

ROI: Region of Interest: A designated region within an image that is the subject of processing (such as hand detection).

CNN: Convolutional Neural Network: This deep learning algorithm is frequently employed for complex gesture detection problems; it is mentioned in related work but is not used in this project.

An application programming interface, or API, is a collection of tools and protocols used to create software programs like OpenCV.

Mathematical Variables

x, y: The virtual canvas's detected hand or fingertips' coordinates.

C: The identified hand region's contour.

A: The contour's area that filters out noise or tiny items.

The HSV color space's hue, saturation, and value components—h, s, and v—ar
used to identify different skin tones.

NOMENCLATURE

The method of recognizing and deciphering hand or body motions in humans as input for controlling apps is known as gesture recognition.

Hand tracking is a computer vision approach that tracks and recognizes hands' movements and positions in real time.

A digital interface that allows users to mimic writing, painting, or sketching is called a virtual canvas.

Computer vision is the branch of artificial intelligence that makes it possible for machines to comprehend and analyze visual information from the outside environment.

Color segmentation is a technique that separates particular color areas in a picture; it is frequently applied to the detection of hands based on skin tones.

Thresholding: A binarization method that makes object detection easier by dividing an image into black-and-white areas according to pixel intensity.

Acronyms

- *IoT*: Internet of Things
- *CV*: Computer Vision
- *HSV*: Hue, Saturation, Value (Color Space)
- *FPS*: Frames Per Second
- *GUI*: Graphical User Interface
- *AR*: Augmented Reality
- *RGB*: Red, Green, Blue (Color Model)

CONTENT

Table of Contents

ACKNOWLEDGEMENT

The full semester Internship/ Project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology & Science** to allow me to continue my disciplinary/interdisciplinary Internship/ Project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would also like to extend my gratitude to the entire team at **Organization's Name**, especially **Dr. Soumyajit Ghosh** , for their collaboration and support during my time with the organization. I am grateful for the opportunity to work alongside such talented individuals and to learn from their expertise and experiences.

I would sincerely like to thank my department, **Centre for Internet of Things,** for allowing me to explore this project. I humbly thank **Dr. Praveen Bansal**, Assistant Professor and Coordinator, Centre for Internet of Things, for his continued support during the course of this engagement, which eased the process and formalities involved. I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Soumyajit Ghosh**, Assistant Professor, and Centre for Internet of Things, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Divyanshu Rathore

0901IO221033

Centre for Internet of Things

Executive Summary (For Internship)/

Abstract (For Projects)

The Virtual Paint Brush is an innovative program created to allow natural and interactive painting interactions through the use of live hand gesture detection. Conventional digital art instruments depend on input tools such as a mouse or stylus, which frequently restrict accessibility and creativity. This project overcomes these restrictions by introducing a hands-free system that utilizes computer vision methods to recognize hand gestures and interpret them as painting actions.

This application offers real-time recognition of gestures, allows for dynamic color selection from a virtual palette, provides adjustable brush sizes, and offers gesture-based controls for advanced interactions such as erasing and resetting the canvas. This project demonstrates the use of computer vision algorithms in real-world scenarios with Python and OpenCV, providing precision, user-friendliness, and flexibility.

The findings emphasize the possibility of gesture-driven interfaces in digital artistic expression. This project has the potential to be expanded to include augmented reality (AR) and multi-user collaboration, which would make it appropriate for educational and professional settings.

PLAGIARISM CHECK CERTIFICATE

This is to certify that I/we, a student of B.Tech. in **Internet of Things (IOT)** have checked my complete report entitled **"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 15%, which is within the specified limit (20%).

The full plagiarism report along with the summary is enclosed.

Divyanshu Rathore

0901IO221033

Checked & Approved By:

Soumyajit Ghosh 20/11/24

Dr. Soumyajit Ghosh
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

DECLARATION BY THE CANDIDATE

I hereby declare that the work entitled **"Virtual Paint Brush"** is my work, conducted under the supervision of **Dr. soumyajit Ghosh , 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.

Divyanshu Rathore
09011O221033
B.Tech. V Sem

Date: 19/11/24
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:
Soumyajit Ghosh 20/11/24
Dr. Soumyajit Ghosh
Assistant Professor
Centre of IOT
MITS, Gwalior

Departmental Project Coordinator

Dr. Nookala Venu
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

Approved by HoD
20/11/2024
Dr. Praveen Bansal
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

VIRTUAL PAINT BRUSH

Minor Project Report

Submitted for the partial fulfillment of the degree of

Bachelor of Technology

In

Internet of Things (IOT)

Submitted By

Divyanshu Rathore
0901IO221033

UNDER THE SUPERVISION AND GUIDANCE OF

Dr. Soumyajit Ghosh
Assistant Professor

Centre for Internet of Things

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.), INDIA
माधव प्रौद्योगिकी एवं विज्ञान संस्थान, ग्वालियर (म.प्र.), भारत
Deemed university
NAAC ACCREDITED WITH A++ GRADEA

June 2024