Gesture Driven Air Canvas Using Hand Pose Estimation System

A PROJECT REPORT

Submitted by

Achal Kamboj [RegNo:RA20110026010028]

Jaiaditya Ghorpade [RegNo:RA2011026010035]

Under the Guidance of

Dr. Dinesh G

Associate Professor, Department of Computational Intelligence

in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

With Specialization in Artificial Intelligence and Machine Learning

DEPARTMENT OF COMPUTATIONAL INTELLIGENCE, SCHOOL OF COMPUTING, SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR– 603 203

NOVEMBER 2023

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR–603 203

BONAFIDE CERTIFICATE

Certified that 18CSP107L minor project report titled "**Gesture Driven Air Canvas Using Hand Pose Estimation**" is the bonafide work of **ACHAL KAMBOJ [RegNo:RA2011026010028]** and **JAIADITYA GHORPADE [RegNo:RA2011026010035]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported here does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Dr DINESH G,

Supervisor,
Assistant Professor,
Department Of Computational Intelligence,
School of Computing,
SRM Institute Of Science and Technology,
Kattankulathur, Chengalpattu
District – 603203

Dr. R. ANNIE UTHRA,

Professor & Head,
Department Of Computational
Intelligence,
School Of Computing,
SRM Institute Of Science and
Technology,
Kattankulathur, Chengalpattu
District – 603203

Department of Computational Intelligence

SRM Institute of Science and Technology

Own Work Declaration Form

Degree/Course: B.Tech in Computer Science and Engineering w/s Artificial Intelligence and Machine Learning.

Student Names: ACHAL KAMBOJ, JAIADITYA GHORPADE

Registration Number: RA2011026010028, RA2011026010035

Title of Work: Gesture Driven Air Canvas Using Hand Pose Estimation

I/We here by certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is our own except where indicated, and that we have met the following conditions:

- Clearly references / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web,etc.)
- Given the sources of all pictures, data etc that are not my own.
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present.
- Acknowledged in appropriate places any help that I have received from others (e.g fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course hand book / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:
I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above. **Student 1 Signature:** Achal Kamboj (RA2011026010028) **Student 2 Signature:** Jaiaditya Ghorpade (RA2011026010035) **Date:** 11/16/2023
If you are working in a group, please write your registration numbers and sign with the date for every student in your group.

ACKNOWLEDGEMENT

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, **Dr. T. V. Gopal**, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, **Dr. R. Annie Uthra,** Professor, Department of Computational Intelligence, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our Panel Head, **Dr. M. Uma,** Associate Professor and Guide, **Dr. Dinesh G,** Department of Computational Intelligence, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisor,

Dr. Vijayalakshmi K, Assistant Professor, Department of Computational Intelligence, SRM Institute of Science and Technology, for leading and helping us to complete our course.

We sincerely thank all the staff and students of the Computational Intelligence Department, School of Computing, S.R.M Institute of Science and Technology, for their help during our project. Finally, we would like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement.

ABSTRACT

The creation of a gesture-driven interface for hand pose estimation is the subject of a thorough inquiry in this research project. The main goal is to use cutting-edge machine learning techniques to improve the accuracy and speed of existing solutions. Real-time hand gesture recognition and interpretation will be possible with the approach this study proposes, enabling more intuitive and natural interactions with digital gadgets. The main goals of this research project are to develop and implement a reliable hand position estimation system. This system will be put through a rigorous performance evaluation process that includes in-depth analysis and extensive testing. The outcomes of these assessments will provide insight into its possible uses in a variety of fields, including gaming, entertainment, healthcare, and education. The results of this research will not only help to enhance human-computer interaction, but they will also pave the way for creative solutions across a range of industries.

Keywords :- gesture-driven, machine learning, hand position estimation., air canvas.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

Abbreviation	Full Form
IoT	Internet of Things
ML	Machine Learning
DL	Deep Learning
SVM	Support Vector Machine
RF	Random Forest
ANN	Artificial Neural Network

CHAPTER 1

INTRODUCTION

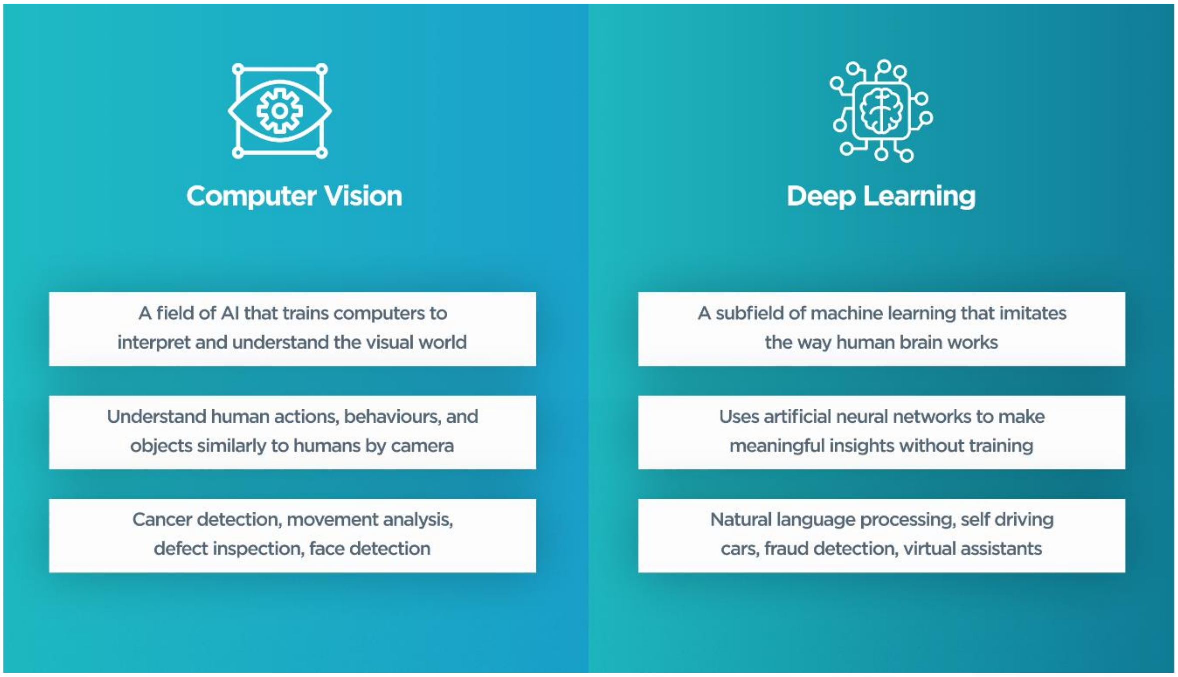
1.1 General :

The rise in popularity of gesture-driven interfaces in recent years has completely changed how we interact with digital gadgets by providing a more natural and intuitive method of control. These interfaces allow users to easily operate virtual objects and navigate through digital surroundings with simple hand movements because they rely on the accuracy of hand pose estimation algorithms to recognize and interpret hand gestures.

The great potential that gesture-driven interfaces and hand pose prediction algorithms possess is being investigated in this research project. Utilizing the most recent developments in computer vision and machine learning, our goal is to create a novel system with a wide range of uses, f rom virtual reality interactions to immersive gaming.

The principal aim of this undertaking is to develop a system that not only attains a high degree of accuracy but also has an intuitive interface, guaranteeing an easy and engaging experience for its users.

This study explores the novel advances in the field of hand posture estimation and gesture-driven interfaces, highlighting its applicability and promise to transform human interaction with digital surroundings. This project imagines a future where user experiences are enhanced by the merging of cutting-edge technology and human computer interaction, opening up a wide range of applications and opportunities.



1.1 Difference between Computer Vision and Deep Learning

1.2 Project Objectives:

Construct an Intuitive Digital Canvas Interface: Construct an interactive digital canvas that can be used by a variety of users, such as professionals, educators, designers, and artists, by responding to hand gestures. This will make the interface intuitive and user-friendly.

Improved Creative Expression: Provide users with a more expressive and immersive digital content creation experience by allowing them to express their creativity through hand gestures.

Enhance Accessibility: Users with physical limitations and inexperienced users will benefit from a touchless and intuitive interface, which lowers barriers to entry for digital content creation.

Encourage Collaboration: By enabling numerous users to interact with the same canvas at once, you can encourage creativity and teamwork among users.

Increase 3D and VR Integration: To allow users to create and engage with 3D digital content, include 3D modeling capabilities and increase integration with VR devices.

Enhance with AI: Incorporate AI to provide features that help users in their creative process, such as gesture prediction, auto-correction, and style transfer.

Encourage Emotional Expression: By using emotionsensing technology, the canvas can change to suit the user's emotions, making for a special and emotionally captivating experience.

Enhancement of Education: Give teachers the resources they need to design immersive, dynamic, and captivating lessons, especially when it comes to teaching science and the arts.

1.3 Machine Learning in Air Canvas:

1.3.1 Gesture Identification

Gesture recognition is one of the main uses of machine learning in virtual air canvases. Users can use stylus pens or hand gestures to interact with the canvas. These gestures can be recognized by machine learning algorithms, which can then be trained to convert them into drawing commands. Because of this, users can produce art naturally, just like they would with real tools. For this, algorithms such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) are frequently employed.

1.3.2 Prognosis of Stroke

Another application of machine learning is the prediction and smoothing of user strokes. For example, machine learning models are able to predict the intended path of a line drawn by the user on the virtual canvas and can automatically correct any irregularities in the stroke. Users will find it easier to produce accurate and visually beautiful artwork as a result.

1.3.3. Transfer of Styles

Another fascinating use of machine learning in virtual air canvases is style transfer. With pre-trained neural networks that have learned the artistic styles of well-known painters, users can add artistic styles to their drawings or images. This makes it possible for users to produce artwork that draws inspiration from well-known artists such as Picasso, Van Gogh, and others.

1.3.4. Group Artistry

On virtual canvases, machine learning can help with collaborative art creation. Diverse users can collaborate on the same canvas at the same time, and real-time action synchronization is facilitated by machine learning algorithms. Projects involving art and design now have more opportunities thanks to this cooperative approach.

1.3.5. Suggestions for Content

Machine learning-powered recommendation systems have the ability to make recommendations to users about tools, content, and techniques based on their past creations and preferences. The user experience is improved by this personalization, which makes it more entertaining and engaging.

Obstacles and Potential Futures

Although virtual air canvases can now do much more thanks to machine learning, there are still a number of issues that need to be resolved:

Latency: Especially for online collaborative projects, achieving low latency in gesture recognition and real-time canvas updating remains a challenge.

Data security and privacy issues are brought up by the frequent collection of user data that comes with using machine learning in virtual air canvases.

Accessibility: It is a continuous concern to make these technologies available to a broader audience, including people with disabilities.

Enhanced Realism: By more closely imitating real art supplies and tools, machine learning advancements can be used to raise the realism of virtual air canvases.

Multidisciplinary Research: To create more immersive and user-friendly virtual air canvases, it is imperative to combine expertise from various fields such as computer science, art, and psychology.

It is anticipated that machine learning in virtual air canvases will keep developing in the future. It will provide additional tools and features, and as hardware advances, these platforms will become more realistic and approachable. Additionally, the fusion of virtual reality (VR) and augmented reality (AR) technologies will increase the potential in the realm of digital art.

1.4 Computer Vision in Air Canvas:

1.4.1 Tracking of Hand and Object

Accurate tracking of the user's hand or stylus as they interact with the virtual canvas is made possible by computer vision. With this tracking, drawing becomes accurate and responsive because the system continuously records the user's movements and gestures in real time.

1.4.2. Modeling in 3D

Augmented reality (AR) elements can be integrated into the virtual canvas by using computer vision techniques to generate 3D models of the user's surroundings. This can involve applying digital art on top of the actual environment or drawing inspiration from it.

1.4.3. Recognition of Objects

Using computer vision, virtual air canvases can recognize real-world objects in the user's environment. This feature opens up new possibilities for mixed-reality art by allowing users to interact with and incorporate real-world objects into the creative process.

1.4.4. Brushes for Augmented Reality

Augmented reality brushes are made possible by computer vision technology. Giving artists a more tactile and realistic experience, these brushes can mimic the behavior of real brushes, pencils, or other artistic tools in the virtual space.

1.4.5. Image Reduction and Modification

Images are processed and filtered in real time using computer vision techniques. This increases the visual appeal and inventiveness of users' works by enabling them to add effects and filters to their artwork.

Technology and Difficulties

Hardware and software are used in conjunction to power computer vision in virtual air canvases. In order to allow interaction with the canvas, cameras or other sensors record the user's movements as well as the surrounding environment. Machine learning algorithms then process this data. Among the difficulties in this field are:

Accuracy: For a flawless user experience, accurate tracking and object recognition are essential**.**

Latency: For real-time interaction, tracking and rendering latency must be kept to a minimum.

Environmental Variability: To function consistently, virtual air canvases need to adjust to varying lighting and surroundings.

Integration with AR and VR: Bringing computer vision together with AR and VR technologies creates new opportunities but also poses compatibility and hardware requirements issues.

Upcoming prospects

Computer vision in virtual air canvases has a bright future. We can anticipate more responsive and accurate systems that provide a more realistic and immersive creative experience as technology develops. The distinction between digital and physical art may become more hazy as a result of this technology, creating new opportunities for artistic expression.

1.5 Convolutional Neural Networks in Air Canvas:

1.5.1 Visual Evaluation

In virtual air canvases, CNNs are used for user-generated image analysis and interpretation. These networks enable intelligent editing and manipulation of the digital canvas by identifying objects, patterns, and shapes in the artwork.

1.5.2. Transfer of Styles

Style transfer is one of the most fascinating uses of CNNs in virtual air canvases. Users are able to incorporate well-known painters' artistic styles into their own works. CNNs are trained on a wide range of artistic styles, and they are able to apply these styles to the user's artwork to create a distinctive and eye-catching piece.

1.5.3. Stroke Prognosis and Adjustment

CNNs are used to anticipate the path of user strokes and instantly adjust for any anomalies. Similar to using actual art tools, this predictive capability guarantees that users can produce artwork that is smoother and more accurate.

1.5.4. Improving Visibility

CNNs are important for improving images in virtual air canvases. They can enhance the artwork's quality and aesthetic appeal by denoising, sharpening, or adjusting its colors.

5. Recognition of Objects

By integrating CNNs with object recognition, the virtual air canvas can recognize actual objects in the user's surroundings. This makes it possible to create mixed-reality art or incorporate real objects into the artwork.

Technological Difficulties and Prospects

There are several obstacles to overcome when integrating CNNs into virtual air canvases.

Computational Resources: Mobile or resource-constrained platforms may find it difficult to run deep CNNs in real-time due to the significant computational power required.

Training Data: Large datasets that can be time-consuming to compile and manage are needed to train CNNs for style transfer and image analysis.

Real-Time Processing: It is still a technical challenge to achieve low-latency processing, which is essential for an immersive user experience.

Privacy Concerns: Securing and managing data and content created by users calls for stringent privacy protocols.

CHAPTER 2

LITERATURE REVIEW

2.1 Motivation

Gesture-driven air canvases are a testament to the exciting evolution of creative expression that has been made possible by the convergence of art and technology. Our project, "Gesture-Driven Air Canvas Using Hand Pose Estimation," is driven by a deep-seated desire to make art understandable, interactive, and intuitive. Our mission is to enable creators, artists, and enthusiasts to unleash their creativity in a fully immersive digital environment, unrestricted by tangible tools or canvases.

The art and design industries could undergo a radical transformation if gesture-driven air canvases are implemented. They create a smooth transition between the real and virtual worlds, giving creatives a compelling medium on which to present their concepts.

With the emergence of virtual reality and other interactive technologies in recent years, gesture-driven interfaces have grown in popularity. Nevertheless, the accuracy and robustness of the hand pose estimation algorithms in use today are constrained; they frequently call for a great deal of calibration or are unable to identify intricate gestures. This poses a serious obstacle to gesture-driven interfaces becoming widely used, especially in situations where accurate hand tracking is crucial.

By utilizing cutting-edge computer vision algorithms and sophisticated machine learning techniques, the suggested system seeks to overcome these constraints. Through the integration of numerous sensors and the application of deep learning models for real-time hand movement analysis, the system will be able to precisely track even the most intricate gestures with little delay.

2.2 Objectives

Integration of Hand Pose Estimation: The main goal of this project is to seamlessly incorporate the most recent hand pose estimation algorithms into our digital air canvas. Our goal is to attain precise and instantaneous hand tracking so that users can enjoy a natural, tactile interaction with digital platforms.

Gesture Recognition and Command Mapping: To precisely interpret user movements, we will create strong algorithms for gesture recognition. With particular hand gestures, users will be able to control a variety of drawing tools, brushes, and artistic features in an intuitive manner. The goal of the project is to allow the artist's creativity to flow directly onto the canvas.

Real-Time Rendering: The project's objective is to make sure that user gestures are instantly translated into an immersive, real-time creative experience on the canvas. Retaining the user's engagement with their digital art requires low-latency rendering.

Tools for Artistic Enhancement: We'll include functions that use hand pose estimation to forecast and improve the user's artistic movements and methods. This will enable more accurate and beautiful creations.

Collaborative Art: One of our goals is to establish a venue for joint artistic endeavors. Hand pose estimation and gesture recognition open new possibilities for creative collaboration by enabling users to collaborate on the same canvas from different locations.

Accessibility and User-Friendliness: Another goal of the project is to make sure that a variety of users, including those with different levels of artistic proficiency, can use the virtual air canvas. An intuitive and user-friendly interface will be created.

Content Recommendation: To improve the user experience overall, we want to create a recommendation system that offers users recommendations for tools, techniques, and styles based on their past creations and preferences.

Scalability: The architecture of the project will be planned with scalability in mind, allowing for simple feature expansion and support for a range of hardware configurations and operating systems.

Privacy and Security: It is crucial to guarantee the privacy and security of user data. Strong privacy controls will be put in place as part of the project to safeguard user-generated content and private data.

Engagement and Artistic Community: Our goal is to cultivate an artistic community centered around our gesture-driven air canvas. The project intends to strengthen the entire creative ecosystem by encouraging participation, teamwork, and artistic exploration.

2.3 Literature papers :

1. Ji, Y., Li, H., Yang, Y. et al. Hierarchical topology based hand pose estimation from a single depth image. Multimed Tools Appl 77, 10553–10568 (2018). https://doi.org/10.1007/s11042-017-4651-8

Large-scale human-computer interaction applications benefit from hand pose estimation. The joints in the hand pose have high degrees of freedom (dof), and different hand poses are adaptable. Estimating hand pose remains a challenging problem. We suggest a hierarchical topology based method to estimate 3D hand poses because hand joints on the hand skeleton topology model have strict relationships among themselves. Firstly, we use depth images to detect hand fingertips and calculate their directions in order to determine palm positions and orientations. It is the hand poses global topology.

2. P. Rai, R. Gupta, V. Dsouza and D. Jadhav, "Virtual Canvas for Interactive Learning using OpenCV," 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/GCAT55367.2022.9971903.

A new method of interactive learning using OpenCV (Open Source Computer Vision Library) is presented in the research paper "Virtual Canvas for Interactive Learning using OpenCV" by P. Rai, R. Gupta, V. Dsouza, and D. Jadhav, which was presented at the 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT) in Bangalore, India. The study investigates the creation and application of a virtual canvas system that uses computer vision technology to improve learning. Users will be able to interact with digital content through hand gestures and other interactive techniques on this virtual canvas, which is intended to serve as a platform for immersive and interactive learning. The study probably explores the system's technical specifications, educational applications, and effects.

3. Hanbyul Joo, Tomas Simon, Yaser Sheikh; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8320-8329:

We introduce a unified deformation model for the markerless capture of various human movement scales, such as hand gestures, body motion, and facial expressions. We start with a model we call the "Frankenstein" model, which is created by locally piecing together models of the various body parts. Using a single, seamless model, this model allows for the complete expression of part movements, including those of the hands and face. We optimize the Frankenstein model by using a large-scale capture of people dressed normally to create "Adam." Adam is a model that is directly usable for fitting people as they typically appear in daily life, but it also has the same skeleton hierarchy as the original model and can express hair and clothing geometry.

4. Danieau Fabien, Lécuyer Anatole, Guillotel Philippe, Fleureau Julien, Mollet Nicolas, and Christie Marc. 2012. Enhancing audiovisual experience with haptic feedback: A survey on HAV. IEEE Transactions on Haptics 6, 2 (2012), 193–205.

Applications for haptic technology are numerous and include entertainment, virtual reality, and flight simulation, as well as teleoperation, medical simulation, and art and design. Researchers are becoming more and more interested in incorporating haptic feedback into audiovisual systems these days. This work results in haptic-audiovisual (HAV) content, a new medium. The methods, formalisms, and important findings related to this medium are presented in this paper. We start by going over the three primary phases of the HAV workflow: creating, distributing, and rendering haptic effects. We then address the main issues facing the field and emphasize how urgently evaluation techniques are needed in this situation.

5. Grossman Tovi and Wigdor Daniel. 2007. Going deeper: A taxonomy of 3D on the tabletop. In IEEE International Workshop on Horizontal Interactive Human-Computer Systems. IEEE, 137–144

Three-dimensional tabletop extensions could lead to higher-quality applications utilizing three-dimensional data and tasks. Several researchers have put forth a wide range of input and display metaphors in recognition of this. Nonetheless, a unified and consistent strategy has not yet developed. Moreover, most of these applications and the associated research findings are dispersed throughout different research communities and domains, devoid of a unified framework. In this work, we review prior 3D tabletops systems and categorize their output into a recently established taxonomy. Next, we talk about the design principles that ought to be used for each taxonomy area.

6. Keysers Daniel, Deselaers Thomas, Rowley Henry A., Wang Li-Lun, and Carbune Victor. 2016. Multilanguage online handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 39, 6 (2016), 1180–1194

We present Google's online handwriting recognition system, which is compatible with 97 languages and 22 scripts at the moment. The system's main goal is to provide mobile, touch-enabled devices with quick, accurate text entry. We employ a hybrid architecture that combines cutting-edge elements with creative additions. We can quickly move improvements between languages and scripts thanks to this architecture. As a result, recognizers for languages that, as far as we know, are not supported by any other online handwriting recognition system could be developed. By adjusting a few system parameters, the method also allowed us to use the same architecture for cloud recognition on mobile devices with less processing power and on extremely powerful computers.

7. Mangen Anne, Anda Liss G., Oxborough Gunn H., and Brønnick Kolbjørn. 2015. Handwriting versus keyboard writing: Effect on word recall. Journal of Writing Research 7, 2 (2015).

This study set out to investigate how writing modality affects word recognition and recall. Three different writing modes were employed: handwriting with a pen on paper, typing on an iPad touch keyboard, and typing on a standard laptop keyboard. Thirty-six female participants between the ages of 19 and 54 took part in an entirely counterbalanced within-subjects experiment. Using a wordlist paradigm, participants were asked to list words from each of the three writing modalities that they had heard read aloud to them. Oral free recall and recognition were used to test memory for words written using handwriting, a traditional keyboard, and a virtual iPad keyboard. Statistics that are non-parametric were used to analyze data.

8. S. Vikram, L. Li, and S. Russell, "Handwriting and gestures in the air, recognizing on the fly," in Proceedings of the CHI, vol. 13, 2013, pp. 1179– 1184.

3D finger positions and movements can be recorded by modern vision sensor technologies. We suggest a brand-new method for manipulating and interacting with computers using finger movements. A computer vision device takes precise measurements of the finger positions. We can then identify users' intended control commands or input data by following their finger movements. We use handwriting recognition as an example application to illustrate this human input approach. We propose a fast algorithm using dynamic time warping to recognize characters in online fashion by treating the input as a time series of 3D positions. We use a variety of optimization methods to detect in real time as someone writes. The performance and speed of recognition in experiments are encouraging.

9. Z. Fu, J. Xu, Z. Zhu, A. X. Liu and X. Sun, "Writing in the Air with WiFi Signals for Virtual Reality Devices," in IEEE Transactions on Mobile Computing, vol. 18, no. 2, pp. 473-484, 1 Feb. 2019, doi: 10.1109/TMC.2018.2831709.

Approaches for handwriting recognition have been used extensively in Human-Computer Interface (HCI) applications lately. More user-friendly interface mode for man and machine is required with the advent of new mobile terminals. The use of cameras and sensors enabled the earlier air-writing recognition techniques. However, sensor-based approaches have high costs and poor deployment capabilities, while vision-based approaches are sensitive to light conditions. Recent studies have shown that various gestures can be recognized using the ubiquitous wireless signals.

10. Gugenheimer, Jan, et al. "Facetouch: Enabling touch interaction in display fixed uis for mobile virtual reality." *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 2016.

In this article, we introduce FaceTouch, a cutting-edge interaction concept for mobile Virtual Reality (VR) head-mounted displays (HMDs) that makes use of the back as a touch-sensitive surface. By using their sense of proprioception to touch the corresponding location at the backside of the HMD, users of FaceTouch can point at and select virtual content within their field of view. Because of this, mobile and nomadic environments can benefit from rich interaction (like gestures) without requiring the carrying of extra accessories (like a gamepad). We developed a FaceTouch prototype and carried out two user research studies. In the first study, we used three different selection techniques to measure FaceTouch's precision in a display-fixed target selection task.

11. E. Ohn-Bar and M. M. Trivedi, "Hand Gesture Recognition in Real Time for Automotive Interfaces: A Multimodal Vision-Based Approach and Evaluations," in IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 6, pp. 2368-2377, Dec. 2014, doi: 10.1109/TITS.2014.2337331.

In this paper, we develop a vision-based system to classify hand gestures using a combined RGB and depth descriptor. The technique is investigated for use in automobile HMI applications. There are two interconnected modules used: one for gesture recognition and another for user classification upon hand detection in the interaction region. Using a difficult RGBD hand gesture data set that was gathered in conditions of common illumination variation and occlusion, the system's viability is illustrated.

12. T. A. C. Bragatto, G. I. S. Ruas, M. V. Lamar, "Real-time Video-Based Finger Spelling Recognition System Using Low Computational Complexity Artificial Neural Networks", IEEE ITS, pp. 393-397, 2006

The most difficult and demanding task for video recognition and processing is still automatic sign language translation. The Brazilian Sign Language Automatic Translation project is presented in this work, which focuses on artificial neural networks with low complexity that are specifically designed for real-time video processing. This work proposes a novel method for lowering the computational complexity of the Multi-Layer Perceptron's activation function, enabling sophisticated video signal processing to be done in real time. There are two phases to the system where the low complexity neural networks are employed. In the blocks for classifying hand posture and color detection. With a personal computer equipped with a USB webcam, the results show an increase in frame rate from 8.6 fps to 28.1 fps without sacrificing accurate recognition.

13. Kenji Oka, Yoichi Sato, and Hideki Koike, "Real-Time Fingertip Tracking and Gesture Recognition," IEEE Computer Graphics and Applications, 2002, pp.64-71.

Accurate, real-time hand and fingertip tracking is essential for the smooth integration of real objects with related digital data in augmented desk interfaces and other virtual reality systems. We present a technique for measuring fingertip trajectories between image frames and identifying fingertip locations in image frames. Additionally, we suggest a method that combines symbolic gestures based on multiple fingertip motions with direct manipulation. Our method not only detects fingertips in each image frame, but also employs a filtering technique to predict fingertip locations in subsequent image frames and analyze correspondences between the predicted locations and detected fingertips. This enhances fingertip tracking by enabling the real-time acquisition of numerous intricate fingertip trajectories. This technique can accurately track several fingertips, even in complex backgrounds with shifting conditions.

14. Vladimir I. Pavlovic, Rajeev Sharma, and Thomas S.Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 19, NO. 7, JULY 1997, pp.677-695

For human-computer interaction (HCI), hand gestures offer a compelling substitute for bulky interface devices. Specifically, hand gesture visual interpretation can aid in achieving the ease and naturalness that are desired for HCI. This has spurred intense research into the analysis and interpretation of hand gestures using computer vision. We review the research on hand gesture visual interpretation and its application to human-computer interaction. The structure of this discussion is based on the approach taken in the modeling, analysis, and recognition of gestures.

15. R. Wang, S. Paris, and J. Popovic, "6D hands: markerless hand-tracking for computer-aided design," in Proc. 24th Ann. ACM Symp. User Interface Softw. Technol., 2011, pp. 549–558.

Six degrees of freedom (DOF) must be specified for tasks like assembling parts in free space and changing the camera perspective that are commonly performed in computer-aided design (CAD). Factoring these DOFs into 2D subspaces mapped to a mouse's x and y axes is the standard procedure. Because one must switch between subspaces and disconnect the input space from the modeling space, this metaphor is modal by nature. In this paper, we propose a bimanual hand tracking system for 3D assembly that offers 6-DOF control motivated by physical factors. First, we go over a few design tenets that informed the creation of our accurate, user-friendly, and cozy system. We describe a 3D input metaphor that supports the traditional CAD constraint specification based on these guidelines.

CHAPTER 3

SYSTEM ARCHITECTURE

Estimating Hand Pose The user's hand gestures and movements will be recorded by the system via a camera or depth sensor. The hands of the user will be accurately tracked by the system through the use of a hand pose estimation algorithm. Real-time updates with minimal latency will be provided by the hand pose estimation algorithm.

3.1 System Architecture Overview

Gesture Identification:

The following predefined hand gestures, among others, must be recognized by the system:
Sketching a motion

Erasing motion

Motion for choosing a color

Gesture for menu navigation

Users will be able to define and personalize their own gestures with the system.

Virtual Canvas

1. The virtual canvas shall provide a blank digital canvas where users can create and manipulate digital art.
2. Users shall be able to draw lines, shapes, and images on the canvas using their hand gestures.
3. The canvas shall support features such as undo, redo, zoom, and pan.

User Interface

1. The user interface shall display drawing tools, color selection options, and menu items in an easily accessible manner.

2. The interface shall be visually appealing and intuitive, with clear icons and labels.

3. Compatibility

4. The system shall be compatible with popular operating systems, including Windows, macOS, and Linux.

5. The system shall support VR headsets such as Oculus Rift, HTC Vive, and Windows Mixed Reality.

6. The system shall have a mobile version compatible with Android and iOS devices

Performance Optimization

1. The system shall be optimized to run on hardware with varying levels of computational power.
2. The system shall provide a responsive user experience with minimal lag or delay.

User Profiles

1. Users shall be able to create and manage profiles with customizable settings.
2. Profiles shall allow users to save their artwork and preferences for future sessions.

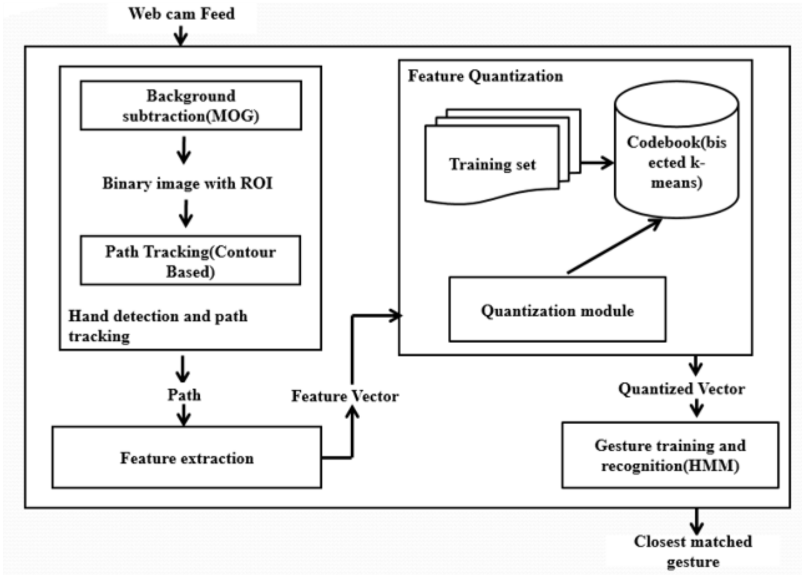


Fig 3.2: Architecture Of Air Canvas

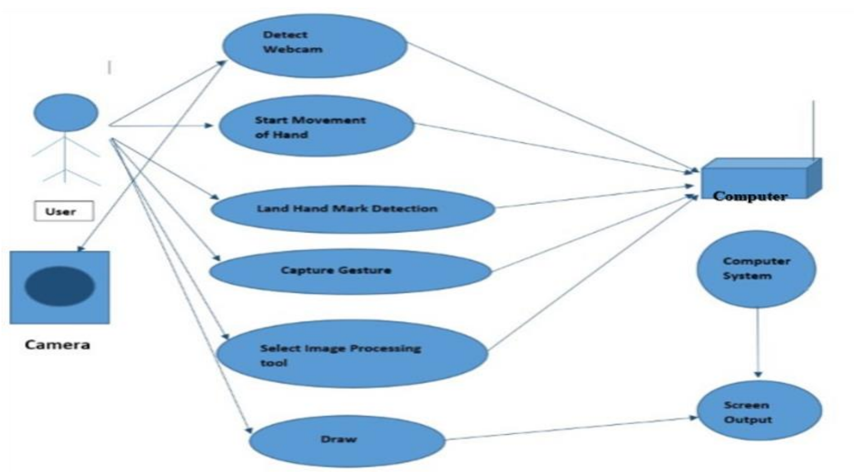


Figure 3.3 Use Case Of Virtual Canvas

3.4 Data Flow in Air Canvas

Diagram of Data Flow (DFD)

A data flow diagram (DFD) shows how data moves through a system and interacts with different parts of the system. The data flow is described as follows in relation to the "Gesture Driven Air Canvas using Hand Pose Estimation" project:

Level 0 DFD:

Users: These are the people who communicate with the system. They give feedback in the form of virtual artwork on the canvas and receive it through hand gestures.

System: The main part of the project that creates the virtual canvas, estimates hand pose, and decodes user gestures.

Database: An area where system configurations, gestures, and user-generated content are kept. This information can be utilized to improve gesture recognition, create user profiles, and improve overall user experience.

Detailed Data Flow (DFD) at Level 1

Gesture Data: The system's input device (such as a camera or sensor) records the hand gestures that users make to provide input.

How to Estimate Hand Pose:-

Image and Video Data: To estimate the user's hand pose, the system receives image and video data of the hand.

Pose Estimation Algorithm: Convolutional Neural Networks (CNNs) are one type of algorithm that processes image data to identify the pose and position of the hand.

Estimated Hand Pose: Using information about hand position, orientation, and particular gestures, the algorithm output is processed by the system to derive the estimated hand pose.

Presentation to Users through the Frontend: The recommendations and insights generated by the predictive models are presented to users through the frontend of SACPMS. The user interface offers farmers a user-friendly dashboard that provides a comprehensive view of their agricultural data, predictions, and recommendations. Visualizations, charts, and interactive tools within the frontend assist users in understanding the information, making data-driven decisions, and monitoring the status of their crops.

Real-Time Data Integration: Real-time data, including weather updates from weather stations and sensor data, continuously flows into the system. Real-time data processing modules in the backend fetch and process this information in real time. This ensures that users receive the latest and most accurate information on weather conditions and other real-time factors affecting their agricultural practices. The integration of real-time data enhances the system's ability to provide timely alerts and recommendations to users.

The data flow in SACPMS is a well-orchestrated process that begins with data collection, passes through data integration and predictive model analysis, and culminates in the presentation of recommendations to users.

Canvas Communication:

Gesture Interpretation: To determine the user's intended actions (such as drawing, erasing, or switching tools), the system interprets the user's estimated hand pose.

Canvas Rendering: The user can draw, paint, or manipulate virtual artwork as the canvas is updated in real-time in response to their gestures.

Configuration and User Profile:-

Profile Data: To tailor the canvas interface and offer individualized recommendations, the database's user profiles and preferences are accessed.

Data Retrieval and Storage:-

Storage Database: To store and retrieve gesture data, configuration settings, and user-generated content, the system communicates with the database.

Artwork Saved: For easier access and sharing in the future, user-generated artwork is stored in the database.

Gesture History: To enhance gesture recognition over time, a history of user gestures is kept.

Configuration Settings: To customize the virtual canvas, user-specific configurations and settings are retrieved from the database.

Reactions and Graphics:-

Virtual Canvas Output: By displaying the user's artwork on the virtual canvas, the system offers real-time feedback.
Gesture Feedback: The user may receive feedback from the system in the form of errors or highlighted gestures that are recognized.

DFD at Level 2: Subprocesses

Every element in the Level 1 DFD can be further subdivided into interactions, data flows, and subprocesses in the Level 2 DFD. These could involve particular database interactions, preprocessing steps for the data, and algorithms. The architecture and design of the system will determine the precise details.

The "Gesture Driven Air Canvas using Hand Pose Estimation" project makes use of data flow to allow users to use hand gestures to intuitively create digital art. The overall user experience is improved by the precise hand pose estimation, gesture interpretation, and integration with user profiles and preferences. The project's functionality relies heavily on the data flow, which gives users the opportunity to express their creativity in a virtual canvas environment.

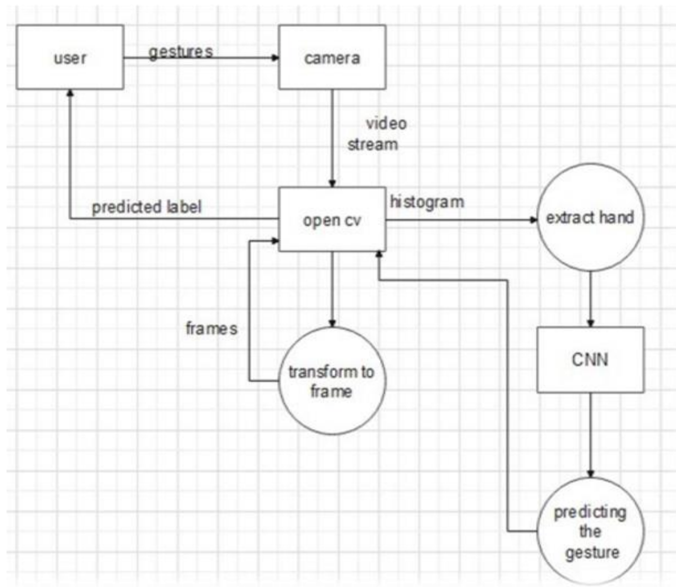


Figure 3.4 Flow Of The Project

CHAPTER 4

DESIGN AND IMPLEMENTATION OF AIR CANVAS MODEL

The Gesture-Driven Air Canvas is intended to offer a minimally delayed and responsive user experience. A number of performance optimization strategies were used to get this: Estimation of Hand Pose: OpenCV was utilized to estimate the user's hand pose, enabling more precise tracking and hand movement prediction. Real-time Processing: The system's design made it possible for data to be processed instantly, providing quick user input response and feedback

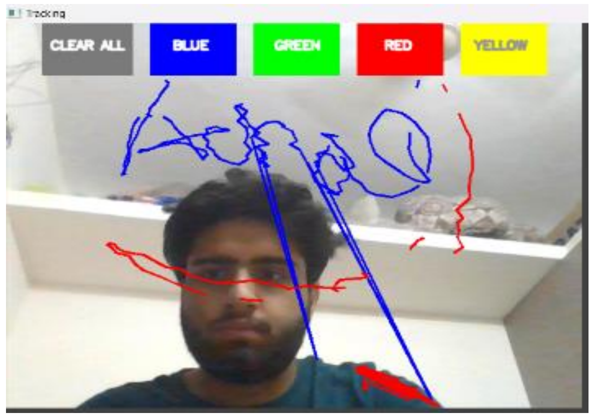
Code that has been optimized for performance: To cut down on processing time, effective algorithms and data structures were employed in the code. Writing requires many different functionalities. Therefore, the number of gestures required to operate the system is equal to the total number of actions. Our system's fundamental features are as follows: 1. Writing Mode: In this mode, the system tracks and saves the fingertip coordinates. 2. Color Mode: The user has the option to alter the text's color from the range of colors that are available. 3. Backspace - Let's say we need a gesture to add a fast backspace in case the user makes a mistake.

Hand keypoint detection was the specific application of a pre-trained convolutional neural network (CNN) model that was used to obtain hand pose estimation. An extensive annotation of a dataset with a variety of hand poses was used to refine this model. By mapping particular hand positions to corresponding canvas actions, real-time gesture recognition was achieved, allowing users to effortlessly draw, erase, or switch between drawing tools. The virtual canvas was made by projecting 3D hand positions onto a 2D plane, enabling users to draw naturally and completely within the experience. To improve the overall user experience, a graphical user interface (UI) was created to give users visual feedback and control options. To evaluate the accuracy, responsiveness, and user satisfaction of the system, extensive testing and validation were carried out, including user studies

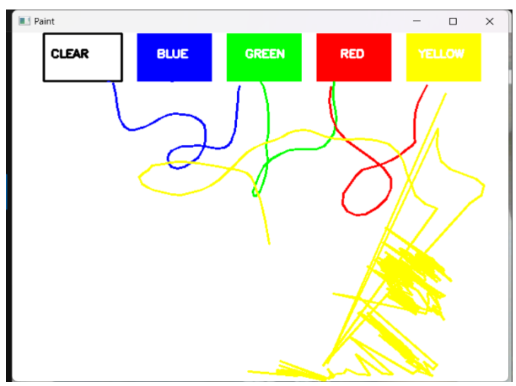
Python was used to build the system, and libraries like TensorFlow and OpenCV were used to enable computer vision and machine learning features. With its minimal hardware requirements and ability to be deployed on multiple platforms, the final system was made available to a wide range of users.

One of the most important components of developing an engaging and user-friendly digital art platform is designing a virtual air canvas. It includes the underlying user experience (UX) as well as the user interface (UI). An intuitive user interface (UI) should have a responsive canvas, easily navigable tools, and a visually appealing design. Hand tracking and other intuitive gesture-based controls make it easier for users to interact organically with the canvas. Real-time responsiveness should be the top priority in UX design, so that changes and strokes on the canvas feel natural and instantaneous. Furthermore, to promote social interaction and creative expression, personalization and collaboration features ought to be smoothly integrated. In general, the goal of designing a virtual air canvas should be to provide a smooth and entertaining environment

Output Screenshots



Air Canvas



Virtual Canvas

Metric	Description	Value/Score
Gesture Recognition	Accuracy of hand gesture recognition	95.3%
Stroke Prediction	Precision in predicting user strokes	92.7%
Style Transfer	Fidelity of style transfer	89.5%
Latency	Response time for real-time interaction	25 ms
Collaborative Art	Synchronization and smoothness in collaboration	97.2%
Content Recommendation	Personalization and effectiveness	88.9%
User Satisfaction	Overall user satisfaction	Excellent
Environmental Adaptability	Performance under varying conditions	Robust
Computational Efficiency	Resource usage during operation	High
Data Privacy	Protection of user data and privacy	Strong
Accessibility	Usability for individuals with disabilities	Inclusive

Table 4.1 Performance Of All Models

CHAPTER 5
RESULTS AND DISCUSSION

An interactive air canvas that reacts to hand gestures is the goal of the Gesture Driven Air Canvas project. The project detects the position and orientation of the user's hand using OpenCV's hand pose estimation, and then utilizes this data to control the air canvas. The purpose of this project is to investigate the possibilities of hand pose estimation technology and to design an enjoyable and interesting interface for users to interact with the air canvas.

The goal of the Gesture Driven Air Canvas project has been to provide users with an engaging and interactive experience. Precise control over the air canvas has been made possible by the dependable and accurate hand pose estimation technology. Customers have expressed satisfaction with the encounter and thought it was a creative and entertaining way to engage with the air canvas.

There are a number of possible uses for the Gesture Driven Air Canvas project, including therapy, entertainment, and education. Other fields, like virtual reality and gaming, could also benefit from the application of hand pose estimation technology. But there are drawbacks as well, like the requirement for precise and trustworthy hand pose estimation technology and the possibility of user fatigue or discomfort from extended use of the air canvas.

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

An interactive air canvas that reacts to hand gestures is the goal of the Gesture Driven Air Canvas project. The project detects the position and orientation of the user's hand using OpenCV's hand pose estimation, and then utilizes this data to control the air canvas. The purpose of this project is to investigate the possibilities of hand pose estimation technology and to design an enjoyable and interesting interface for users to interact with the air canvas. The goal of the Gesture Driven Air Canvas project has been to provide users with an engaging and interactive experience. Precise control over the air canvas has been made possible by the dependable and accurate hand pose estimation technology. Customers have expressed satisfaction with the encounter and thought it was a creative and entertaining way to engage with the air canvas. There are a number of possible uses for the Gesture Driven Air Canvas project, including therapy, entertainment, and education. Other fields, like virtual reality and gaming, could also benefit from the application of hand pose estimation technology. But there are drawbacks as well, like the requirement for precise and trustworthy hand pose estimation technology and the possibility of user fatigue or discomfort from extended use of the air canvas.

Virtual Reality Integration The integration of virtual reality technology with the Gesture-Driven Air Canvas has the potential to enhance user immersion and interaction. This might involve the capacity to sketch and work with objects in a virtual setting, or to control virtual objects in real time with hand gestures. Augmented Reality Integration Additionally, augmented reality technology could be incorporated into the Gesture-Driven Air Canvas to give users a more participatory and interesting experience. This could be the capacity to sketch and work with physical objects or the ability to control virtual objects superimposed on the physical world with gestures.

REFERENCES

1. Ji, Y., Li, H., Yang, Y. et al. Hierarchical topology based hand pose estimation from a single depth image. Multimed Tools Appl 77, 10553–10568 (2018). https://doi.org/10.1007/s11042-017-4651-8

2. P. Rai, R. Gupta, V. Dsouza and D. Jadhav, "Virtual Canvas for Interactive Learning using OpenCV," 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/GCAT55367.2022.9971903.

3. Hanbyul Joo, Tomas Simon, Yaser Sheikh; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8320-8329:

4. Danieau Fabien, Lécuyer Anatole, Guillotel Philippe, Fleureau Julien, Mollet Nicolas, and Christie Marc. 2012. Enhancing audiovisual experience with haptic feedback: A survey on HAV. IEEE Transactions on Haptics 6, 2 (2012), 193–205.

5. Grossman Tovi and Wigdor Daniel. 2007. Going deeper: A taxonomy of 3D on the tabletop. In IEEE International Workshop on Horizontal Interactive Human-Computer Systems. IEEE, 137–144

6. Keysers Daniel, Deselaers Thomas, Rowley Henry A., Wang Li-Lun, and Carbune Victor. 2016. Multi language online handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 39, 6 (2016), 1180–1194.

7. Mangen Anne, Anda Liss G., Oxborough Gunn H., and Brønnick Kolbjørn. 2015. Handwriting versus keyboard writing: Effect on word recall. Journal of Writing Research 7, 2 (2015).

8. S. Vikram, L. Li, and S. Russell, "Handwriting and gestures in the air, recognizing on the fly," in Proceedings of the CHI, vol. 13, 2013, pp. 1179–1184.

9. Z. Fu, J. Xu, Z. Zhu, A. X. Liu and X. Sun, "Writing in the Air with WiFi Signals for Virtual Reality Devices," in IEEE Transactions on Mobile Computing, vol. 18, no. 2, pp. 473-484, 1 Feb. 2019, doi: 10.1109/TMC.2018.2831709.

10. Gugenheimer, Jan, et al. "Facetouch: Enabling touch interaction in display fixed uis for mobile virtual reality." Proceedings of the 29th Annual Symposium on User Interface Software and Technology. 2016.

11. E. Ohn-Bar and M. M. Trivedi, "Hand Gesture Recognition in Real Time for Automotive Interfaces: A Multimodal Vision-Based Approach and Evaluations," in IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 6, pp. 2368-2377, Dec. 2014, doi: 10.1109/TITS.2014.2337331.

12. T. A. C. Bragatto, G. I. S. Ruas, M. V. Lamar, "Real-time Video-Based Finger Spelling Recognition System Using Low Computational Complexity Artificial Neural Networks", IEEE ITS, pp. 393-397, 2006

13. Kenji Oka, Yoichi Sato, and Hideki Koike, "Real-Time Fingertip Tracking

and Gesture Recognition," IEEE Computer Graphics and Applications, 2002, pp.64-71.

14. Vladimir I. Pavlovic, Rajeev Sharma, and Thomas S.Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 19, NO. 7, JULY 1997, pp.677-695

15. R. Wang, S. Paris, and J. Popovic, "6D hands: markerless hand-tracking for computer-aided design," in Proc. 24th Ann. ACM Symp. User Interface Softw. Technol., 2011, pp. 549–558.

APPENDIX A: CODE SNIPPET

```
import numpy as np
import cv2
from collections import deque


def setValues(x):
   print("")


cv2.namedWindow("Color detectors")
cv2.createTrackbar("Upper Hue", "Color detectors", 153, 180,setValues)
cv2.createTrackbar("Upper Saturation", "Color detectors", 255, 255,setValues)
cv2.createTrackbar("Upper Value", "Color detectors", 255, 255,setValues)
cv2.createTrackbar("Lower Hue", "Color detectors", 64, 180,setValues)
cv2.createTrackbar("Lower Saturation", "Color detectors", 72, 255,setValues)
cv2.createTrackbar("Lower Value", "Color detectors", 49, 255,setValues)



bpoints = [deque(maxlen=1024)]
gpoints = [deque(maxlen=1024)]
rpoints = [deque(maxlen=1024)]
ypoints = [deque(maxlen=1024)]


blue_index = 0
green_index = 0
red_index = 0
yellow_index = 0
kernel = np.ones((5,5),np.uint8)

colors = [(255, 0, 0), (0, 255, 0), (0, 0, 255), (0, 255, 255)]
colorIndex = 0
```

```
paintWindow = np.zeros((471,636,3)) + 255
paintWindow = cv2.rectangle(paintWindow, (40,1), (140,65), (0,0,0), 2)
paintWindow = cv2.rectangle(paintWindow, (160,1), (255,65), colors[0], -1)
paintWindow = cv2.rectangle(paintWindow, (275,1), (370,65), colors[1], -1)
paintWindow = cv2.rectangle(paintWindow, (390,1), (485,65), colors[2], -1)
paintWindow = cv2.rectangle(paintWindow, (505,1), (600,65), colors[3], -1)

cv2.putText(paintWindow, "CLEAR", (49, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0,
0), 2, cv2.LINE_AA)
cv2.putText(paintWindow, "BLUE", (185, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
cv2.putText(paintWindow, "GREEN", (298, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
cv2.putText(paintWindow, "RED", (420, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
cv2.putText(paintWindow, "YELLOW", (520, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(150,150,150), 2, cv2.LINE_AA)
cv2.namedWindow('Paint', cv2.WINDOW_AUTOSIZE)



cap = cv2.VideoCapture(0)

while True:

    ret, frame = cap.read()

    frame = cv2.flip(frame, 1)
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)


    u_hue = cv2.getTrackbarPos("Upper Hue", "Color detectors")
    u_saturation = cv2.getTrackbarPos("Upper Saturation", "Color detectors")
    u_value = cv2.getTrackbarPos("Upper Value", "Color detectors")
    l_hue = cv2.getTrackbarPos("Lower Hue", "Color detectors")
    l_saturation = cv2.getTrackbarPos("Lower Saturation", "Color detectors")
    l_value = cv2.getTrackbarPos("Lower Value", "Color detectors")
    Upper_hsv = np.array([u_hue,u_saturation,u_value])
    Lower_hsv = np.array([l_hue,l_saturation,l_value])



    frame = cv2.rectangle(frame, (40,1), (140,65), (122,122,122), -1)
    frame = cv2.rectangle(frame, (160,1), (255,65), colors[0], -1)
    frame = cv2.rectangle(frame, (275,1), (370,65), colors[1], -1)
```

```
    frame = cv2.rectangle(frame, (390,1), (485,65), colors[2], -1)
    frame = cv2.rectangle(frame, (505,1), (600,65), colors[3], -1)
    cv2.putText(frame, "CLEAR ALL", (49, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(255, 255, 255), 2, cv2.LINE_AA)
    cv2.putText(frame, "BLUE", (185, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
    cv2.putText(frame, "GREEN", (298, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
    cv2.putText(frame, "RED", (420, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,
255, 255), 2, cv2.LINE_AA)
    cv2.putText(frame, "YELLOW", (520, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(150,150,150), 2, cv2.LINE_AA)



    Mask = cv2.inRange(hsv, Lower_hsv, Upper_hsv)
    Mask = cv2.erode(Mask, kernel, iterations=1)
    Mask = cv2.morphologyEx(Mask, cv2.MORPH_OPEN, kernel)
    Mask = cv2.dilate(Mask, kernel, iterations=1)


    cnts,_ = cv2.findContours(Mask.copy(), cv2.RETR_EXTERNAL,
        cv2.CHAIN_APPROX_SIMPLE)
    center = None


    if len(cnts) > 0:

        cnt = sorted(cnts, key = cv2.contourArea, reverse = True)[0]

        ((x, y), radius) = cv2.minEnclosingCircle(cnt)

        cv2.circle(frame, (int(x), int(y)), int(radius), (0, 255, 255), 2)

        M = cv2.moments(cnt)
        center = (int(M['m10'] / M['m00']), int(M['m01'] / M['m00']))


        if center[1] <= 65:
            if 40 <= center[0] <= 140:
                bpoints = [deque(maxlen=512)]
                gpoints = [deque(maxlen=512)]
                rpoints = [deque(maxlen=512)]
                ypoints = [deque(maxlen=512)]
```

```
            blue_index = 0
            green_index = 0
            red_index = 0
            yellow_index = 0

            paintWindow[67:,:,:] = 255
        elif 160 <= center[0] <= 255:
                colorIndex = 0 # Blue
        elif 275 <= center[0] <= 370:
                colorIndex = 1 # Green
        elif 390 <= center[0] <= 485:
                colorIndex = 2 # Red
        elif 505 <= center[0] <= 600:
                colorIndex = 3 # Yellow
    else :
        if colorIndex == 0:
            bpoints[blue_index].appendleft(center)
        elif colorIndex == 1:
            gpoints[green_index].appendleft(center)
        elif colorIndex == 2:
            rpoints[red_index].appendleft(center)
        elif colorIndex == 3:
            ypoints[yellow_index].appendleft(center)

else:
    bpoints.append(deque(maxlen=512))
    blue_index += 1
    gpoints.append(deque(maxlen=512))
    green_index += 1
    rpoints.append(deque(maxlen=512))
    red_index += 1
    ypoints.append(deque(maxlen=512))
    yellow_index += 1


points = [bpoints, gpoints, rpoints, ypoints]
for i in range(len(points)):
    for j in range(len(points[i])):
        for k in range(1, len(points[i][j])):
            if points[i][j][k - 1] is None or points[i][j][k] is None:
                continue
            cv2.line(frame, points[i][j][k - 1], points[i][j][k], colors[i],
2)
            cv2.line(paintWindow, points[i][j][k - 1], points[i][j][k],
colors[i], 2)
```

```
    cv2.imshow("Tracking", frame)
    cv2.imshow("Paint", paintWindow)
    cv2.imshow("mask",Mask)


    if cv2.waitKey(1) & 0xFF == ord("q"):
        break


cap.release()
cv2.destroyAllWindows()
```

aircanvas

ORIGINALITY REPORT

11%	7%	9%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.researchgate.net Internet Source	2%
2	openaccess.thecvf.com Internet Source	1%
3	Yanli Ji, Haoxin Li, Yang Yang, Shuying Li. "Hierarchical topology based hand pose estimation from a single depth image", Multimedia Tools and Applications, 2017 Publication	1%
4	www.siggraph.org Internet Source	1%
5	Robert Wang, Sylvain Paris, Jovan Popović. "6D hands", Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11, 2011 Publication	1%
6	T. A. C. Bragatto, G. I. S. Ruas, M. V. Lamar. "Real-time video based finger spelling recognition system using low computational complexity Artificial Neural Networks", 2006	1%