An AI – Based Virtual Drawing System Using Python

Gayathri G Murali[#1], Shukhaira Beegam T[#2],Akhil Raju[#3] ,A Amal[#4], Tessy Abraham Azhikakathu[#5]

[#1#2#3#4]*Student, Department of Computer Science and Engineering, Mahaguru Institute Of Technology,APJ Abdul Kalam Technological University, Alappuzha , Kerala*

[#5]*Faculty, Department of Computer Science and Engineering, Mahaguru Institute Of Technology,APJ Abdul Kalam Technological University, Alappuzha ,Kerala*

Email: [#1]itsgaayume@gmail.com,[#2]shukhaira2000@gmail.com, [#3]akhilsreeraju@gmail.com,[#4]amalanilmangalam@gmail.com [#5] tessyabraham.a@gmail.com

ABSTRACT

This paper presents a virtual painting system that employs artificial intelligence (AI) techniques to enable users to create digital paintings with natural and intuitive gestures. The system uses a machine learning model trained on a large dataset of paintings to predict the user's intended brushstrokes and color choices based on their hand movements and other input signals. This approach allows for a more expressive and natural way of creating digital art, while also reducing the cognitive load of working with complex software interfaces. The system also includes features such as automatic color suggestions and brush stroke smoothing to further enhance the user experience. Overall, this system demonstrates the potential of AI to revolutionize the way we create digital art and opens up new possibilities for artists and designers.

Keywords – Artificial Intelligence, Air Canvas, Digital Art, Gesture Recognition, Hand Tracking

I. INTRODUCTION

The Virtual AI Painter, which uses OpenCV and Mediapipe, tracks the motion of an object. By moving the item (in our example, the human hand) in front of the webcam while it is in the air, the user can draw on the screen using this tracking capability. The user can design straightforward objects that are both engaging and difficult using this real-time webcam data that is generated by tracking the movement of the object.

OpenCV (Open-Source Computer Vision) is a collection of functions for the programming language that is mostly used for computer vision. It is a library used for image processing, to put it simply or in a generic way. It is mostly used for all procedures involving images.

What it can do is

- Recognize and create images.
- Recognition of faces and their characteristics.
- Recognition of various shapes, including circles, rectangles, and others, in a picture. For instance, finding coins in photographs.
- Image text recognition. Consider reading license plates.
- Has the ability to alter the colour or image quality.
- Making apps for augmented reality. A library for images is called OpenCV.

Almost all popular programming languages are supported. frequently applied in C++ and Python. To read or write an image for image manipulation, use OpenCV. Transform coloured images into grey, binary, HSV, etc. OPENCV is an Open source as well.

Google's open-source framework is called MediaPipe. used for media processing (graph based). primary goals towards simplifying the process of processing media for us by giving characteristics of machine learning and some incorporated computer vision

II. PROBLEM STATEMENT

To create an AI-based tool that uses OpenCV principles and can draw anything on any surface by simply using a camera to capture the action of a coloured marker. The marker in this instance is a colourful item held at the tip of the finger.

III. MOTIVATION

One of the most intriguing and difficult research areas in the world of AI is writing in the air. These were the issues that troubled us, and it was as a result that we came up with the project idea for the air canvas. Everyone wants to sketch or draw in the air with their hands.

Sometimes we picture writing with our hand in the air. So, based on this idea, we created a canvas,

104

chose the necessary colours with our hands, and then sketched or wrote the necessary design on it.

IV. EXISTING SYSTEM

Your screen is a tool that performs just what you want it to do—display information. There are several ways to do this, one of which is to use the keyboard, which is a well-established and popular means to display data on the screen. It features a keyboard made up of buttons used to produce letters, numbers, and symbols as well as execute other operations. The second technique uses speech-to-text software, which listens to audio files and outputs an editable, verbatim transcript on a specific device. Through voice recognition, the software accomplishes this. The disadvantage of this approach is that the voice recognition software won't always translate your words precisely on the screen. Programmes frequently make mistakes that are the result of misinterpretation since they are unable to distinguish between homonyms and cannot comprehend language context the way that humans can. A touchscreen is another approach. It is a computer screen that can be operated without a mouse or keyboard by touching it with a finger or a stylus pen. It can be compared to a touchpad with an integrated screen.

They are unsuitable for entering big amounts of data, are not very accurate, and picking detailed items with fingers can be challenging. It is more expensive than substitutes like a mouse and, if mistreated, can quickly go bad.

V. PROPOSED SYSTEM

● An AI based drawing system is a software which uses webcam to detect and trace hand movements making users comfortable in generating drawings or texts.
● This system does not collect user's personal information and guarantees security.
● Latest technology of AI along with computer vision makes our system more user-engaging.
● Our system is an open source, cheap & always accessible.

VI. LITERATURE REVIEW

Users can draw on and interact with virtual things in the air using interactive technologies called air canvas drawing systems. Numerous industries, such as art, design, education, and entertainment, can benefit from these systems. In this study of the literature, we'll look at a number of research on air canvas drawing methods and their potential effects across a range of industries. In one study, air canvas sketching techniques were used in the field of art, according to Vazquez-Alvarez et al. (2018). The researchers discovered that these technologies made it easier and faster for artists to produce drawings with higher complexity and depth than they could with conventional techniques. The authors also mentioned that air canvas drawing systems can contribute to the democratization of the art world by giving aspiring artists easier access to and more cheap materials. The use of air canvas drawing systems in the classroom has also been investigated. The application of these systems in the classroom, specifically for the teaching of mathematics, was examined in a research by Li et al. (2020). The researchers discovered that air canvas drawing techniques could aid students in better visualizing mathematical ideas and gaining a deeper comprehension of the material. The authors also pointed out that similar platforms may be used to support collaborative learning, enabling students to collaborate on issues and exchange ideas in real-time. Finally, a study by Zhang et al. (2018) shows that air canvas drawing systems have potential uses in the entertainment industry. Users of the interactive game created by the writers could sketch and move virtual objects in the air. Users responded positively to the game, reporting high levels of involvement and enjoyment.

Finally, air canvas drawing systems have the potential to revolutionize a number of industries, including as art, design, education, and entertainment. According to the studies discussed below, these platforms can democratize access to tools and technologies while simultaneously enhancing creativity, cooperation, and learning results. To completely comprehend the effects of these systems and create best practices for their application in various contexts, more research is necessary.

VII. TOOLS AND METHODOLOGY

Anaconda :

Anaconda is an open-source distribution for the Python and R programming languages. It is used for data science, machine learning, deep learning, and so on. With the availability of more than 300 libraries for data science, it is truly ideal for any developer to work with Anaconda for data science.

Hardware Requirements:

Operating System: Windows 11 Processor: CPU N3710 @1.60GHz System Type: 64-bit operating system, x64-basedprocessor Installed Ram: 8 GB GPU: NVIDIA GeForce GTX 800 or higher Web cam (For real-time hand Detection)

Software Requirements:

The software required to run this project is Python. Python is an interpreted, high-level, general-purpose programming language that was created by Guido van Rossum and first released in 1991. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming.

OpenCV:

OpenCV is a library of programming functions mainly aimed at real-time computer vision. The library is cross-platform and free to use under the open-source BSD license. It was originally developed by Intel and later supported by Willow Garage.

NumPy:

NumPy is a package designed for efficient array processing. It provides high-performance multidimensional array objects, and tools for working with these arrays.

VIII. PROPOSED METHODOLOGY

The system has two sides, a back and a front. The back is made up of three pieces: a camera, a detector, and an interface.

Fig 8.1:Data flow diagram

> Camera module:

Input module comprises of elements answerable for interfacing and catching input signs from various types of picture markers and sending it to the detecting module to be deals with it as of frames. The ordinarilyutilized strategies for catching and recognizing input are hand maneuvers gear for computerization and recording, and still or motion cameras. In our system, we use an inner web-camera which is helpful and cost- conscious addition to see both inactive and moving pictures.
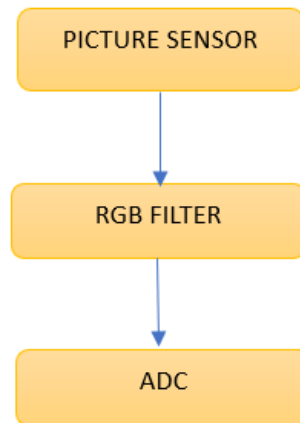

Fig 8.2: Web camera module

> Detection module:

This module is solely responsible for image processing. The output from a camera module is then passed through to different image processing methods, for example; color conversion, noise removal and thresholding. After these processes have been completed the image goes through contour extraction. If the image contains defects, Convexity defects are then found and used to identify the gesture[6][7]. However, if there are no defects present in the image then it is classified using Haar Cascade to identify the gesture[1].
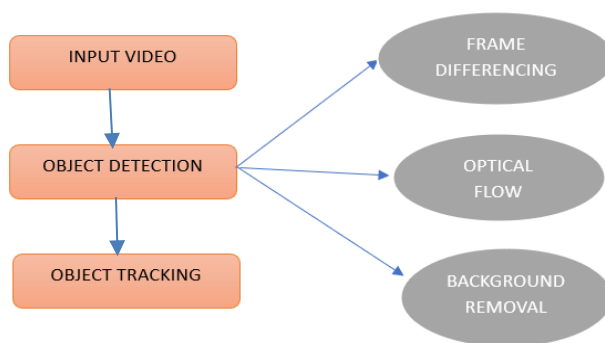

Fig 8.3: Object detection module

106

➢ Interface Module:

The rear end computes the video gestures that are being made and maps them to their associated recognized classical hand gestures. The purpose of this module is to take the gestures detected by the camera and calibrate them to the corresponding actions. These actions are then sent to the appropriate application. The front end of the module consists of three windows. The main window shows the video input from the camera along with the name of the gesture being identified. The second window displays the contours found inside the primary image. The third window presents the smooth thresholded adaptation of the original image. The back end of the module calculated the video gestures being made and assigns them to the corresponding recognized classical hand gestures. It adopts image processing techniques[9][10].
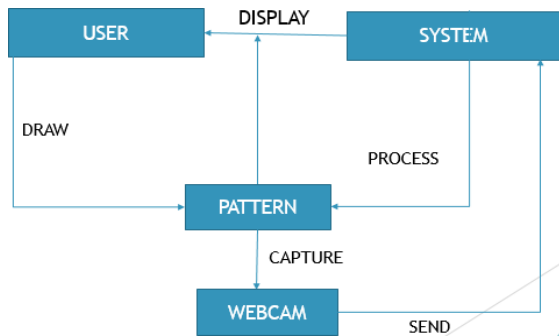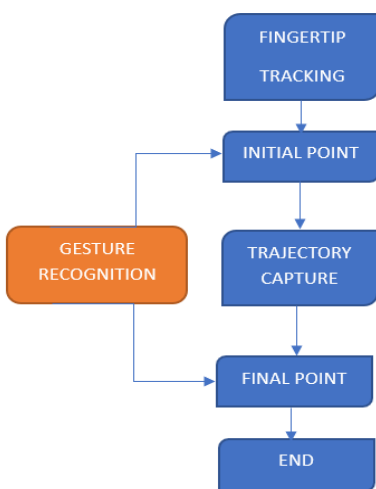

Fig 8.4: Interface module

➢ Hand Tracking Module

Fig 8.5: Hand tracking module

A computer can recognize a hand from an input image using computer vision, and hand tracking[5] is theprocess of keeping an eye on the hand's movement and orientation. We can create a wide range of

programmes that employ the direction and movement of the hand as their input thanks to hand tracking. We frequently use the same code to provide hand tracking as part of our programme across many projects. This issue is resolved by building a hand tracking module because we just need to write the code once. After that, we turn this section of code into a module. This module may be imported into any Python project we are working on, and it will carry out hand tracking.

1. Palm detection: MediaPipe uses the entire input image to produce a cropped hand image.

2. Identifying hand landmarks: Using a cropped image of the hand, MediaPipe identifies 21 hand landmarks.

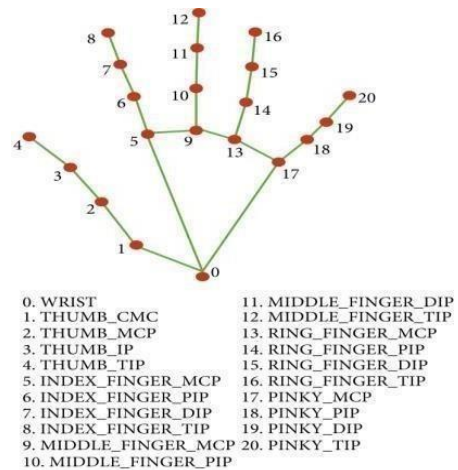The figure below depicts the 21 hand points that MediaPipe recognises:

Fig 8.6 : Hand Landmarks Detection

IX. RESULT AND DISCUSSION

Python, OpenCV, and other python modules are the foundation of this project. We learn how to choose an element and use that choice to initiate an action. fresh ideas and techniques for solving problems. When developing the hand tracking module, we learn how to use OpenCV concepts like limiting the active hand detection and assigning the action to the fingers. We acquire the project deployment techniques. We use modern applications like PyCharm, Visual Studio Code, and Figma. We have the opportunity to deal with several platforms,

107

languages, and technologies by working on this project. This project introduces us to further fresh Python modules and technologies.

We acquire accustomed to selecting the component and starting the action with this particular decision. Error-correcting strategy and novel concepts. When it came time to create the hand trackingmodule, we figured out how to put OpenCV concepts like assigning the activity to the fingers and limiting the dynamic hand placement into practice.

Using these functions and algorithms we can draw Circles, Rectangle, Ellipse, and Free hand drawings with 3 different colors. The generated pattern could be saved using CTRL+S and file is saved in .png format. So obtained drawing could be shared through social media platforms.

Figure 9.1: Header

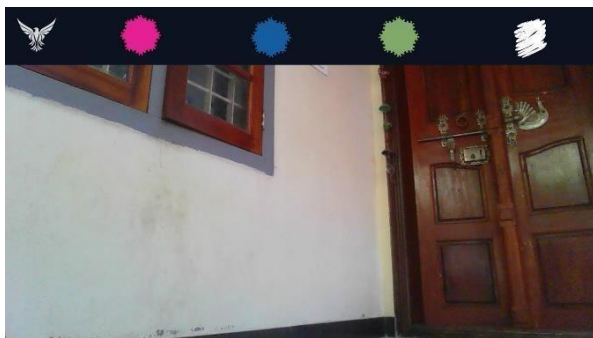The Header has been created using Online editing tool Canva.
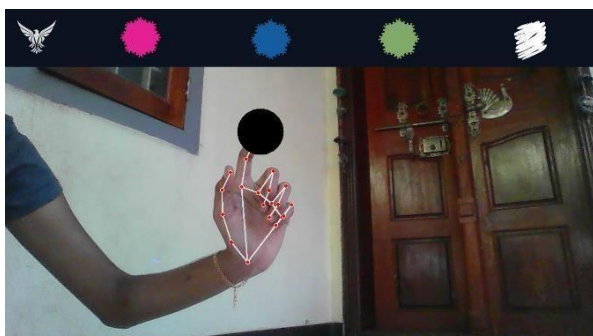

Fig. 9.2 :Interface

Fig 9.3: Hand Tracking

Fig 9.4 :Pattern Generation

X. CONCLUSION

This AI Based virtual painter is capable of employing complex conventional writing techniques. It provides a simple way to take notes, eliminating the need to hold a smart phone in one hand. Theultimate goal is to develop a computer vision device learning application that supports human-computerinteraction (HCI), also known as human-laptop interaction (MMI)[14], which is the relationship between people and computers in general and the device in particular. With thehelp of this project, the client can create an interactive environment in which he or she can draw whatever they desire by selecting their chosen colours from the palette.

XI. FUTURE SCOPE OF THE PROJECT

If we had more time to devote to this endeavor, we would enhance hand contour recognition, investigate our initial Air Canvas objectives, and make an effort to comprehend the multicore module. We would need to go further into OpenCV in order to improve hand gesture tracking. There are other ways to analyze contours, but for this particular procedure, it would be beneficial to look at the color histogram that was used to draw the contours in question.

Additionally, we could test out various interpolation techniques. PyGame has a line drawing technique (pygame.draw.line ()) that might be helpful for creating lines that are smoother and cleaner. In the same line, adding different brush types, textures, and perhaps a rubber to Air Canvas will strengthen its artistic capabilities. Unique features that imitate actual creativity software could also include letting

the user save their finished product or watching their drawing process as an animation. There might even be a way to link Air Canvas with real digital drawing applications like Adobe Photoshop, Clip Studio Paint, or GIMP! Finally, by understanding how multicore processing interacts with in-order information processing, we could make significant progress.

- Voice Assistant : Making use of Voice Assistantto navigate the website and identify photos.

- The need for image processing applications increased as a result of the inclusion of cameras in mobile devices such smartphones, iPad, and tablets. The fact that the mobile device is solely powered by a battery means that these applications must be quicker and use less power.

- Robot Control : A system that uses numbering to count the five fingers for controlling a robot via hand position signs has been proposed as one of the fascinating applications in this subject.

- Online Teaching:- This method also supports and encourages online teaching which involves HCI.[2]

XII.   REFERENCES

[1] Vladimir I. Pavlovic, Student Member, IEEE, Rajeev Sharma, Member, IEEE and Thomas S. Huang, Fellow, IEEE "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review" VOL. 19, NO. 7, JULY 1997

[2]Gangadhara Rao Kommu, Assistant ProfessorDepartment Of Information technology, Chaitanya Bharati Institute of Technology, Hyderabad, India "AN EFFICIENT TOOL FOR ONLINE TEACHING USING OPENCV"

[3] PranaviSrungavarapu, Eswar Pavan Maganti, SrilekkhaSakhamuri, Sai Pavan Kalyan Veerada, Anuradha Chinta "Virtual Sketch using Open CV" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075 (Online), Volume-10 Issue-8, June 2021

[4] " AIR CANVAS APPLICATION USING OPENCV AND NUMPY IN PYTHON"International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: 08 | Aug 2021.

[5] Alper Yilmaz, Omar Javed, Mubarak Shah, "Object Tracking: A Survey", ACM Computer Survey. Vol. 38, Issue. 4, Article 13, Pp. 1-45, 2006

[6] https://ieeexplore.ieee.org/document/8270084/authors.

[7] T. Grossman, R. Balakrishnan, G. Kurtenbach, G. Fitzmaurice, and B. Buxton, "Creating Principal 3D Curves with Digital Tape Drawing," Proc. Conf. Human Factors Computing Systems (CHI' 02), pp. 121- 128, 2002.

[8] H.M. Cooper, "Sign Language Recognition: Generalising to More Complex Corpora", Ph.D. Thesis, Centre for Vision, Speech and Signal Processing Faculty of Engineering and Physical Sciences, University of Surrey, UK, 2012

[9] Yusuke Araga, Makoto Shirabayashi, Keishi Kaida, Hiroomi Hikawa, "Real Time Gesture Recognition System Using Posture Classifier and Jordan Recurrent Neural Network", IEEE World Congress on Compu- tational Intelligence, Brisbane, Australia, 2012

[10] EshedOhn-Bar, Mohan ManubhaiTrivedi, "Hand Gesture Recognition In Real-Time For Automotive Interfaces," IEEE Transactions on Intelligent Transportation Systems, VOL. 15, NO. 6, December 2014, pp 2368-2377

109