Synopsis Report on

Virtual Whiteboard-A Gesture Controlled Pen-free Tool

Submitted in partial fulfillment of the requirements of the degree of

Bachelor of Engineering

By

KAMLAKANT BAG (58)

SIDHARTH URANKAR (59)

ANKITA YADAV (60)

Supervisor:

PROF. MONALI SANKHE

Computer Engineering Department

VIVA Institute of Technology

University of Mumbai

2022-2023

CERTIFICATE

This is to certify that the project entitled **"Virtual Whiteboard A Gesture Controlled Pen-free Tool"** is a bonafide work of **"Kamlakant Bag, Siddharth Urankar and Ankita Yadav"** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering in Computer Engineering"**

Prof. Monali Sankhe
Guide

Prof. Ashwini Save
Head of Department

Dr. Arun Kumar
Principal

Project Synopsis Approval for B.E.

This project report entitled **Virtual Whiteboard-A Gesture Controlled Pen-free Tool** by **Kamlakant Bag, Siddharth Urankar, Ankita Yadav** is approved for the degree of **Bachelor of Engineering in Computer Engineering**.

Examiners

1.----------

2.----------

Date:

Place:

Abstract

The computer vision field has been rapidly developing, finding real-world applications, and even surpassing humans in solving some of the visual tasks. All this thanks to the recent advances in artificial intelligence and learning. Object tracking is considered as one the important task within the field of computer vision.

A computing process that attempts to recognize and interpret human gestures using mathematical algorithms is known as gesture recognition. With increasing technology each sector needs to be modernized. With the improvement of clever gadgets, the system can be now controlled virtually with aid of using human gestures. While using paint, sometimes we feel difficult to draw and feel like drawing our imagination just by waving our hand. The proposed system this gap in developing motion-to-text converter which can serve as software for smart wearable devices for writing in the air. The proposed system will use a computer vision to track finger movement. This proposed system works on hand tracking system development which aims to track the hand which acts as pen and functioning as pen to create or draw different shapes and also as an eraser using Open Computer Vision Library (OpenCV) and Media pipe. The existing project which allows us to draw just by waving hand uses technology or methodology which takes a lot of process and time. Avoiding or decreasing these limitations we came up with this proposed system that uses new technologies and easy methodologies. System Camera is used to track the hand and create drawings.

Table of Contents

List of Figures

List of Tables

Declaration

We declare that this written submission represents our ideas in my own words and where others' ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will because for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Kamlakant Bag

Siddharth Urankar

Ankita Yadav

Date:

Chapter 1

Introduction

Earlier painting was done using either a mouse or touch pad which was quite stressful and hectic task. Even though we have touch screen laptops, they are expensive. Hand tracking more specifically finger tracking technique is used as a tool of the computer acting as an external device like a keyboard and a mouse. It is used in various fields like Virtual Reality to sign language recognition. Air Canvas is a hands-free digital drawing canvas which utilizes camera, OpenCV and media pipe to recognize and map the hand gestures. The user finger is considered as the brush, or the pen used to draw or annotate pdf. The size of brush can be modified, also the pen color can be changed by hovering pointer over built-in buttons. Computer vision techniques are used to draw different shapes. This proposed system uses python language to build the code. Camera and Media

pipe is used to track the finger positions. Computer Vision built in methods are used to draw shapes on the canvas or the area provided. We can annotate or edit pdf of our choice by opening the required pdf and hovering over the area where we need to annotate or underline. We can also save the canvas work as image Air canvas helps to draw on a screen just by waiving your finger fitted with a colorful point or a simple colored cap. We will be using the computer vision techniques of OpenCV to build this project. The preferred language is python due to its exhaustive libraries and easy to use syntax but understanding the basics it can be implemented in any OpenCV supported language. The presentation is a source of communication between a speaker and an audience. It conveys information, delivers a lecture, demonstrates new ideas, and exhibits your thoughts to a group of people. It illuminates the content of a topic to a learner, either in academia or for business purposes. PowerPoint is popular software that improves your presentation skills, by providing a visual illustration of your content. It allows presenting text, diagrams, audios, videos, statistical graphs, animations, etc. A presenter can move PowerPoint slides forward or backward by using a mouse, a keyboard, or remote control, but these Human-Computer Interactions (HCI) gadgets are inconvenient and incommodious. However, a hand gesture recognition system has been proposed that will control PowerPoint slides without any disturbance during the presentation.

Chapter 2

Literature survey

The following chapter is a literature survey of the previous research papers and researches which gives detailed information about the previous system along with its advantages and disadvantages.

2.1 Survey of existing system

A survey was done on the existing literature and products to find out their shortcomings and research gaps in their systems. This survey consisted of more than 15 papers wherein the most relevant are listed below.

Rafiqul Zaman Khan, et.al [1] provided a study on Hand Gesture Recognition, in this paper a survey of recent hand gesture recognition systems is presented. Key issues of the hand gesture recognition system are presented with challenges. Review methods of recent postures and gestures recognition system presented as well. Summary of research results of hand gesture methods, databases, and comparison between main gesture recognition phases are also given. Advantages and drawbacks of the discussed systems are explained finally. In this paper various methods are discussed for gesture recognition, these methods include from Neural Network, HMM, fuzzy c-means clustering, besides using orientation histogram for features representation. For dynamic gestures HMM tools are perfect and have shown its efficiency especially for robot control. NNs are used as classifier. In this work application areas for the gestures system are presented. Total Gestures used For Training and Testing is 1040 data used is American sign language recognition percentage is 98.7%, when total gestures used for training and testing is 60 recognition percentage by normal method is 84% and dataset used is Own Database. Drawbacks Neural Network classifier has been applied for gestures classification, but it is time consuming. In System limitations restrict the application such as; gestures are made with the right hand only, the arm must be vertical, the palm is facing the camera, background is uniform.

Zhuihu hu, et.al [2] implemented Gesture detection using RGB hand image using convolutional neural network. In this paper we study the hand gesture recognition in human computer interaction. In order to deploy HRG system on RGB only cameras we propose to use CNN based model to recognize the visual information. Two stages are used to first detect the hand region and then to recognize the iconic gesture. Compared to the available conventional machine learning models our proposed method has an advantage in recognition accuracy. The HRG based system may innovate new applications in natural human computer interaction and improve the user experience in consumer electronics, intelligent manufacturing, and video games. In our experiment we verified our gesture recognition system on two databases. We can see that the detection rate reaches above 90% when threshold is 25%. For removing the background image for hand ROI, the accuracy is satisfactory. State-of-the-art end-to-end CNN is used for comparison with dropout ratio of 0.3. We

can see that the recognition ratio reaches 81.2% which is better than other state-of-the-art recognition algorithms.

Pinar Kirci, et.al [3] have proposed Hand Gesture Detection, in this study, by using computer vision and image processing techniques, with interpreting the hand movements of the user that enter the vision field of the camera, it can be used more functionally and the user can manage the computer without any physical contact and as being away from the monitor. In programming, version 3.7.1 of the Python platform was used. A certain area size is created in a part of the image and the hand movement is read through this area. We have defined a frame area to place our hand. Then we have obtained a black and white image using the Otsu Method. The areas are determined in the manner of covering the fingers (Contour Drawing) and number of fingers are determined over the area. Then the gaps between the two fields are determined. The convexity Defects method is used while gaps are found. It predicts the meaning of the hand motion that is done by fingers with utilizing the indicated gaps between the fingers and makes the appropriate processes that are expected from the finger motions.

Ishika Dahl, et.al [4] demonstrated Automated Hand Gesture Recognition using a Deep convolutional neural network. This paper discusses and offers a state-of-the-art deep Multi- layer Convolutional Neural Network for performing hand gesture recognition in Human-Robot Interaction systems. It is an efficient model to be used on image data when tuned properly and with proper image pre-processing. The experimental results of this paper show that the model proposed in this paper can distinguish among several dominant and low-level features for the input images and can classify various hand gestures with greater accuracy and a negligible model loss of 0.0504. Before actual detection, the images to be trained are segmented. Afterwards, features are extracted and reduced (finding the significant features and removing the unnecessary details that may create noise or reduce accuracy). The model was able to achieve a testing accuracy of 99.13% which was much higher than the existing models. Gaining an accuracy as high as that was a tough job but it can be achieved by turning various hyperparameter and by performing data preprocessing and

argumentation techniques. The resultant Deep Convolution Network model comprises of seven hidden layers and tuned Hyperparameters for our application.

Sumedh Bansode, et.al [5] have implemented Computer Vision Based Virtual Sketch Using Detection Developing an interface between human palm and the system using open cv techniques and python language to pick the tool and draw using hand on the developed drawing area. For making teaching videos more explanatory. The system has the potential to challenge traditional writing/teaching methods. The goal is to create a computer vision machine learning application that promotes Human computer interaction also named Man-Machine Interaction refers to the relation between the human and the computer or more precisely the machine. System functionality referred to the set of functions or services that the system equip is to the users while system can operate and perform specific user purposes activity efficiently such as virtual drawing. In our approach, we attempted to create a prototype tool that may be used as an alternative to such software. Our tool would be highly useful and would improve online learning. And, because it is cost-effective, it might be used by any teacher to make their teaching films more informative. We also sought to make it as simple and user-friendly as possible, with the very minimum of hardware requirements, so that even someone with no prior computer experience could use it. library used are CV2, media pipe, NumPy, Time, pyttsx3.

Yash Patil, et.al [6] proposed Virtual Painting with OpenCV Using Python This paint application is created using OpenCV module and python programming language which is an apex machine learning tool to create an application like this. Given the real time webcam data, this paint-like python application uses OpenCV library to track an object-of-interest (a bottle cap in this case) and allows the user to draw by moving the object, which makes it both awesome and challenging to draw simple things. The proposed system can be classified into mainly two steps after acquiring the input image from camera, videos or even an Object of Interest. These steps are: Extraction Method image pre-processing and Features. In this machine learning application, we have developed code by using Python programming language along with OpenCV library. Main idea

behind this algorithm is to use live feed from camera and process each frame. However, the algorithms will be implemented on defined ROI (region of interest).
estimation and Extraction.

Niharika M, et.al [7] have proposed Virtual hand gesture recognition system. The virtual paint application's fundamental goal is to deliver an AI-based tool that allows users to draw anything on screen using hand movements. This system also gives the user the option of selecting any tool from the toolbar. The user can save their completed work or see their drawing process as a replay animation with this application. In this paper, we introduce a virtual paint application that uses hand gestures for real-time drawing or sketching on the canvas. Hand gesture-based paint application can be implemented using cameras to capture hand movement. To accomplish activities like as tool selection, writing on the canvas, and clearing the canvas, an intangible interface is created and implemented using vision-based real-time dynamic hand gestures. The images of the hands are taken with the system's web camera and processed in real time with a single-shot detector model and media pipe, allowing the machine to communicate with its user in a fraction of a second.

Ashfaq Ahmad, et.al [8] implemented Controlling PowerPoint Using Hand Gestures in Python The research has tried to control different operations of the PowerPoint slideshow through gestures. This research has used Machine Learning to detect gestures with subtle differences and tried to map them with some fundamental PowerPoint slideshow controlling functions using Python. While presenting, it is difficult to give the perfect presentation because of focusing on multiple things, including the slides, the keys to change the slides, and the audience, while maintaining composure. This research focused on removing one such distraction by allowing the presenter to manage slides solely by gesturing in front of the camera. We managed to map specific gestures for one action on the slides, including the next slide, previous slide, zoom in and out, opening a highlighter/pen, and play/pause videos. This project uses the python language, and the library used for machine learning is Py Torch. We used a dataset named 20BN-jester Dataset V1 for the project, which had 148,092 gestures. The results achieved were excellent, which can make hand gestures a righteous mode of presenting.

Pranavi Sugawara, et.al [9] demonstrated Virtual Sketch using Open CV, in this project we are performing the morphological operations are a set of operations that process images based on shapes. These apply a structuring element to an input image and generate an output image. Developing an interface between human hand and the system using open cv techniques and python language to pick the colour and draw using hand on the developed drawing area. we see the implementation of the project where a colored marker is used by holding it in the hand and four colors are shown respectively as blue, green, red and yellow as well as we have a clear button which is used to erase the drawing done. This project makes the user to have an interactive environment where the user can draw whatever he wants by choosing his required colors from the displayed ones. So, we conclude that Virtual Sketch is developed using the library NumPy and in Open CV where we have many libraries and algorithm in built which makes the interfaces more active while using. We used python as, it have many inbuilt libraries and many modules which represent the imagination virtually when used along with OpenCV as well as its morphological processes.

Zhenmin YUAN, et.al [10] implemented Sketch Recognition based Intelligent Whiteboard Teaching System, this paper realizes an intelligent whiteboard system based on sketch recognition. The system is built on multi-tier distributed architecture, which can support the users to communicate with each other by sketching and chatting on Pen-based HCI. It can recognize the gestures to help users modify, copy and move sketches as familiar pencil-and-paper process, and transform the free-hand sketches into the symbols of the domain-specific knowledge automatically using feature points detection and graphics recognition algorithms. Such novel AI-aid sketching communication can support building connection between the concrete graphics and the abstract concepts during the learning activities, especially for information technology education. The experimental whiteboard system fulfills a flowchart-to-C programming and an ER diagram-to-database recognition and communication framework. The result of the usability evaluation of the experiment suggests that this system has higher efficiency for collaborative learning and more sufficient for the concept learning of information science. The paper is organized as follows. In

Section 2 we give a brief overview of the intelligent whiteboard system, and provide a general function description of the system. Section 3 gives the multi-tier distributed architecture for the E-Learning system. Section 4 presents the details of the gesture and sketch recognition algorithms. Section 5 gives the experiments of applied system and usability evaluation. Finally, some conclusions are drawn.

Table 2.1: Analysis Table

Title	Summary	Advantages	Technology Used
Hand gesture recognition.[1]	In this paper a survey of recent hand gesture recognition systems is presented. Key issues of hand gesture recognition system is presented with challenges of gesture system.	American sign language recognition percentage is 98.7%, when total gestures used for training and testing is 60 recognition percentage by normal method is 84%	Neural Network, HMM, fuzzy c-means clustering
Gesture detection using RGB hand image using convolutional neural network.[2]	In this paper we study the hand gesture recognition in human computer interaction. In order to deploy HRG system on RGB only cameras we propose to use CNN based model to recognize the visual	For removing the background image for hand ROI, the accuracy is satisfactory. State-of-the-art end-to-end CNN is used for comparison with dropout ratio of 0.3. We can see that the	ROI, CNN

	information. Two stages are used to first detect the hand region and then to recognize the iconic gesture	recognition ratio reaches 81.2% which is better than other state-of-the-art recognition algorithms.	
Hand Gesture Detection.[3]	In this paper by using computer vision and image processing techniques interpreting the hand movements of the user that enter the vision field of camera and the user can manage the computer without any physical contact and as being away from the monitor.	Hand motion are monitor and number of fingers are determined over camera.	Otsu method.
Automated Hand Gesture Recognition using a Deep convolutional neural network.[4]	This paper discusses and offers a state-of-the-art deep Multi-layer Convolutional Neural Network for performing hand gesture recognition in Human-Robot	The model was able to achieve a testing accuracy of 99.13%	CNN

	Interaction systems. The experimental results of this paper show that the model proposed in this paper can distinguish among several dominant.		
Computer Vision Based Virtual Sketch Using Detection.[5]	Detection Developing an interface between human palm and the system using open cv techniques and python language to pick the tool and draw using hand on the developed drawing area.	it is cost-effective.	CV2, media pipe, NumPy, Time, pyttsx3.
Virtual Painting with OpenCV Using Python.[6]	This paint application is created using OpenCV module and python programming language which is an apex machine learning tool to create an	created a computer vision machine learning application that promotes Human computer interaction (HCI) also named Man Machine Interaction (MMI)	Open CV

	application like this. The proposed system can be classified into mainly two steps after acquiring the input image from camera, videos or even an Object of Interest. These steps are: Extraction Method image pre-processing and Features estimation and Extraction		
Virtual paint using hand gesture.[7]	This system also gives the user the option of selecting any tool from the toolbar. The user can save their completed work or see their drawing process as a replay animation with this application. In this paper, we introduce a virtual paint application that	The user can save their completed work or see their drawing	Mediapipe, open cv NumPy

	uses hand gestures for real-time drawing or sketching on the canvas. Hand gesture-based paint application can be implemented using cameras to capture hand movement.		
Controlling PowerPoint Using Hand Gestures.[8]	The research has tried to control different operations of the PowerPoint slideshow through gestures. This research has used Machine Learning to detect gestures with subtle differences and tried to map them with some fundamental PowerPoint slideshow controlling functions using Python.	The results achieved were excellent, which can make hand gestures a righteous mode of presenting.	NumPy, OpenCV.
virtual Sketch using Open CV.[9]	Virtual Sketch is in where we draw by	Interface which is very simple and under	NumPy, OpenCV

	capturing the motion of colored marker with camera. The project provides the user an environment where we can draw whatever he wants by choosing colors from the displayed ones.	stable by user. User will be able to draw what he wishes without any interruption	
Sketch recognition based intelligent whiteboard teaching system.[10]	This paper realizes and intelligent white board system-based sketch recognition, which can support the user to communicate with each other by sketching pen based HCI.	It can recognition the gesture to help users modify, copy and move s familiar pencil and paper process, transform the free hand sketches into the symbol of the domain specific knowledge automatically using feature point detection and graphic recognition algorithm	Sketch recognition algorithm

2.2 Research gap

Writing in air has been one of the most fascinating and challenging research areas in field of image processing and pattern recognition in the recent years. Presentation using slideshow is an effective and attractive way to convey information in the digital world which require devices like mouse, keyboard, or laser pointer etc. The disadvantage is one must have prior knowledge about the devices in order to operate them. This paper proposes a new method to control the slides during a presentation using bare hand. The system is mixture of Virtual whiteboard and PowerPoint controlling system using hand gesture. The previous system does not contain assigning tools using hand gesture. In the existing system we can write only cursive but in the proposed system freehand writing is possible.

2.3 Problem statement and Objective

The existing system only works with your fingers and no highlighters, paints, or relatives. Developing an interface between human palm and the system using open cv techniques and python language to pick the tool and draw using hand on the developed drawing area. For making teaching videos more explanatory. PowerPoint presentations sometimes become less lively because either you must use the keyboard to change and operate the slides or use a dedicated gadget to perform these tasks. We aimed to enable people to control the slideshow with the gestures of hands.

2.4 Scope

This program has the potential to challenge traditional writing methods. Eliminates the need to carry a cell phone in hand to take notes, to give an easy way on the go to do the same. It will again work towards a greater purpose in helping especially those who know them to communicate easily. Even adults who find it difficult to use the keyboard can easily use the program. Expanding

functionality, this program can also be used to control IoT devices soon. Air painting can also be made happen. This program will be very good smart clothing software using which people can work better with the digital world. The unpopular reality of taxpayers we see can make the text come alive. Wind-writing programs should listen only to their master's control touch and should not be misled by people all around.

Chapter 3

Proposed system

This chapter gives the overview of the Proposed system.

3.1 Algorithm

The algorithm for the proposed system is as follows:

3.1.1 Virtual whiteboard:

Step 1: Start reading the frames and convert the captured frames to HSV color space. (Easy for color detection)

Step 2: Prepare the canvas frame and put the respective ink buttons on it.

Step 3: Adjust the trackbar values for finding the mask of colored marker.

Step 4: Preprocess the mask with morphological operations. (Erosion and dilation)

Step 5: Detect the contours, find the center coordinates of largest contour and keep storing them in the array for successive frames. (Arrays for drawing points on canvas)

Step 6: Finally draw the points stored in array on the frames and canvas.

3.1.2 PowerPoint controlling algorithm:

Step 1: Select Suitable file

Step 2: Open the File

Step 3: Start Video Streaming

Step 4: Maintain Gesture Array

Step 5: Transform Array

Step 6: Classification

Step 7: Perform Action

3.2 Details of System

3.2.1 Software Requirements

Language: Python

Platform: Visual studio, PyCharm, Jupyter Notebook.

Library:

Hand gesture recognition and tracking are handled by the Media Pip framework, while computer vision is handled by the OpenCV library. To track and recognize hand movements and hand tips.

- **OpenCV** (for image processing and Color HSV)
- **NumPy** (for Use Array)
- **Media Pip** (for Hand Tracking)
- **AutoPy** (for controlling the keyboard and mouse, finding colors and bitmaps on-screen, and displaying alerts)
- **TensorFlow** (for Image recognition and perform the AI functions)

- **PyTorch** (Use for applications such as Computer vision and natural language processing)

3.2.1 Hardware Requirements

1. Processor: intel i5 (10th gen).
2. Ram: 8GB +.
3. Graphic Card: 2GB.

3.3 Design Details

In this section flow diagrams and block diagrams of the system are explained.

3.3.1 System Flow Diagram

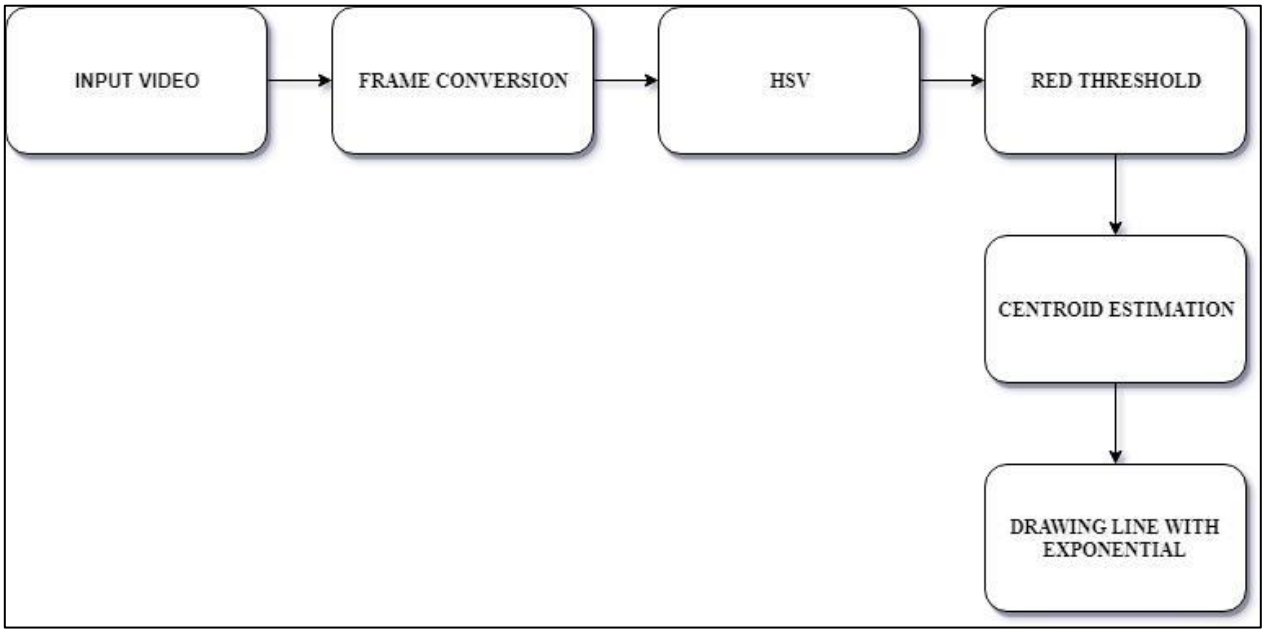


Figure 3.3.1 Flowchart of Virtual Whiteboard

Video is captured with the use of camera and it is converted to frames, by applying the HSV to the frames and the red threshold is applied to the video converted frames.by the use of red threshold centroid value can be estimated. Now we can draw the line using this centroid value with its exponential values.

3.3.2 Block Diagram

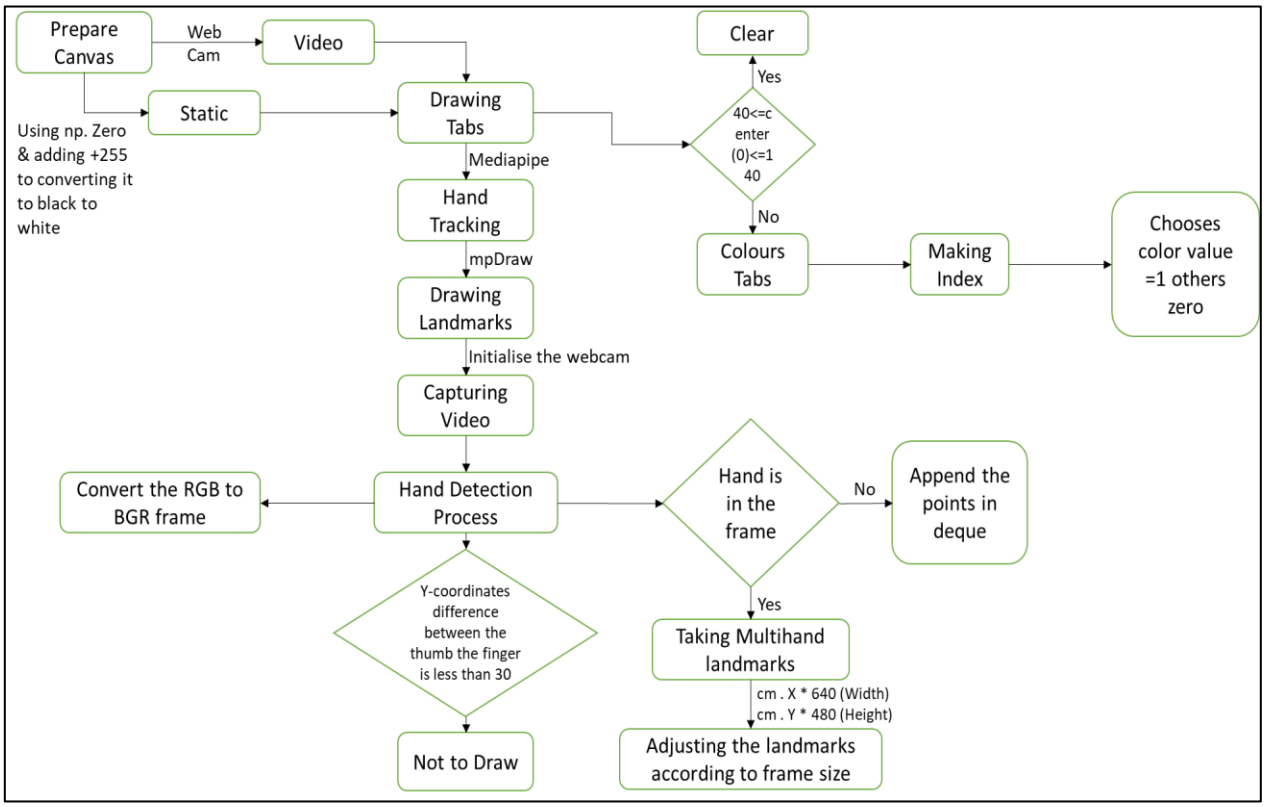


Figure 3.3.2 Block Diagram of Virtual Whiteboard

The above diagram can be viewed in three parts preparing canvas, drawing tabs and hand tracking. Canvas is prepared using NumPy. And hand tracking is done through landmark detection. here 21 landmarks are detected.

3.4 Methodology

The objective is to create a free space where one can draw in air freely. The RGB camera detects the fingertip and tracks its motion throughout the screen. Whenever the hand comes in front of the camera, the initial thing to do is detect the fingertip. There are various ways of fingertip detection.

3.4.1 Fingertip Detection:

We are aiming to develop a system which can accurately detect the fingertips. First, we will detect the whole hand and then the region segmentation is done. Region segmentation is a two-step approach which includes skin segmentation and background subtraction

3.4.2 Fingertip Tracking:

After successive detection of hand region and center of gravity the next step is to track the fingertip movement on the screen. According to the previous work, faster R-CNN handheld detector is intensive and the frames produced are below real-time performance. Thus, we are aiming to use KCF tracting algorithm. The algorithm converts the detected fingertip into HSV color space. After the mask is detected in the air, the system will do some morphological operation to remove the impurities from the masked image. After detecting the contours drawing the line is the most important step. After this a python deque is created. The deque will memorize the position of the outline in each subsequent frame, and we will use these accumulated points to create a line using OpenCV's drawing capabilities

The methodologies or the stages of the proposed system are discussed below:

1. Run or Execute the Code

Execute the code once all the libraries are installed, this leads to turning the camera on automatically and the OpenCV frame with buttons displaying various shapes, colors, size, save, clear, erase etc.

2. Webcam Starts

Webcam starts recording and converts the video each frame and sends the frame to hand tracker class to track or detect the positions of finger. Figure 2 displays the buttons and frame which records video.

3. Detects Hand Landmarks

Each frame received is compared with mediapipe hand landmarks

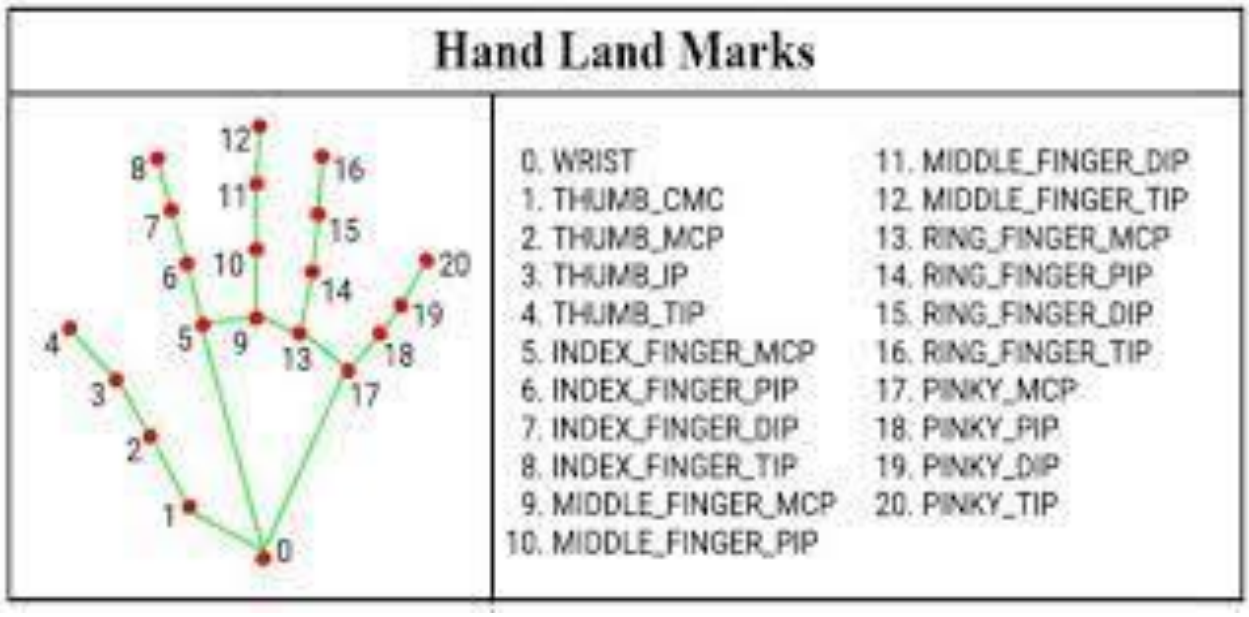


Figure 3.4.1 Landmark Detection Table

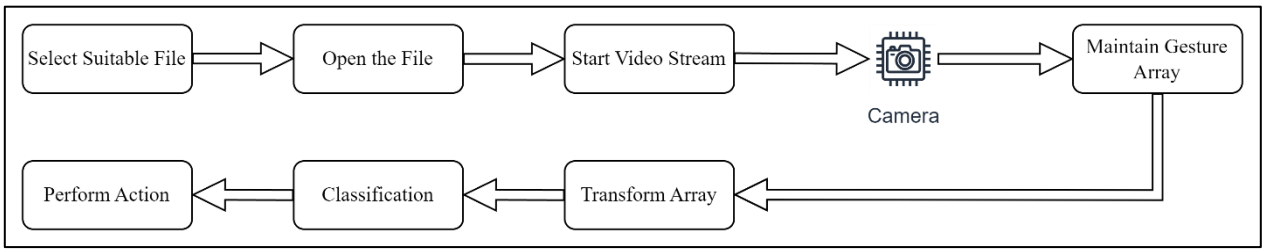


Figure 3.4.2 Step Diagram

The first and second phases include selecting and opening a PowerPoint file for presentation on PowerPoint Windows. In the third phase, the system starts a live video stream for detecting and recognizing the live gestures. The gesturing will be recorded as an image array of size 20 in the fourth phase. The fifth phase transforms this array. In the sixth phase, the array of images is classified using the pre-trained weights, loaded earlier by the system.

3.5 UML Diagram

3.5.1 Use Case Diagram

Use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases involved. Figure 3.3 Use case diagram. Shows use class diagram for the model, it starts up webcam and inputs are taken using hand detections then these inputs are given proper frame rates and according to those gestures perform actions.

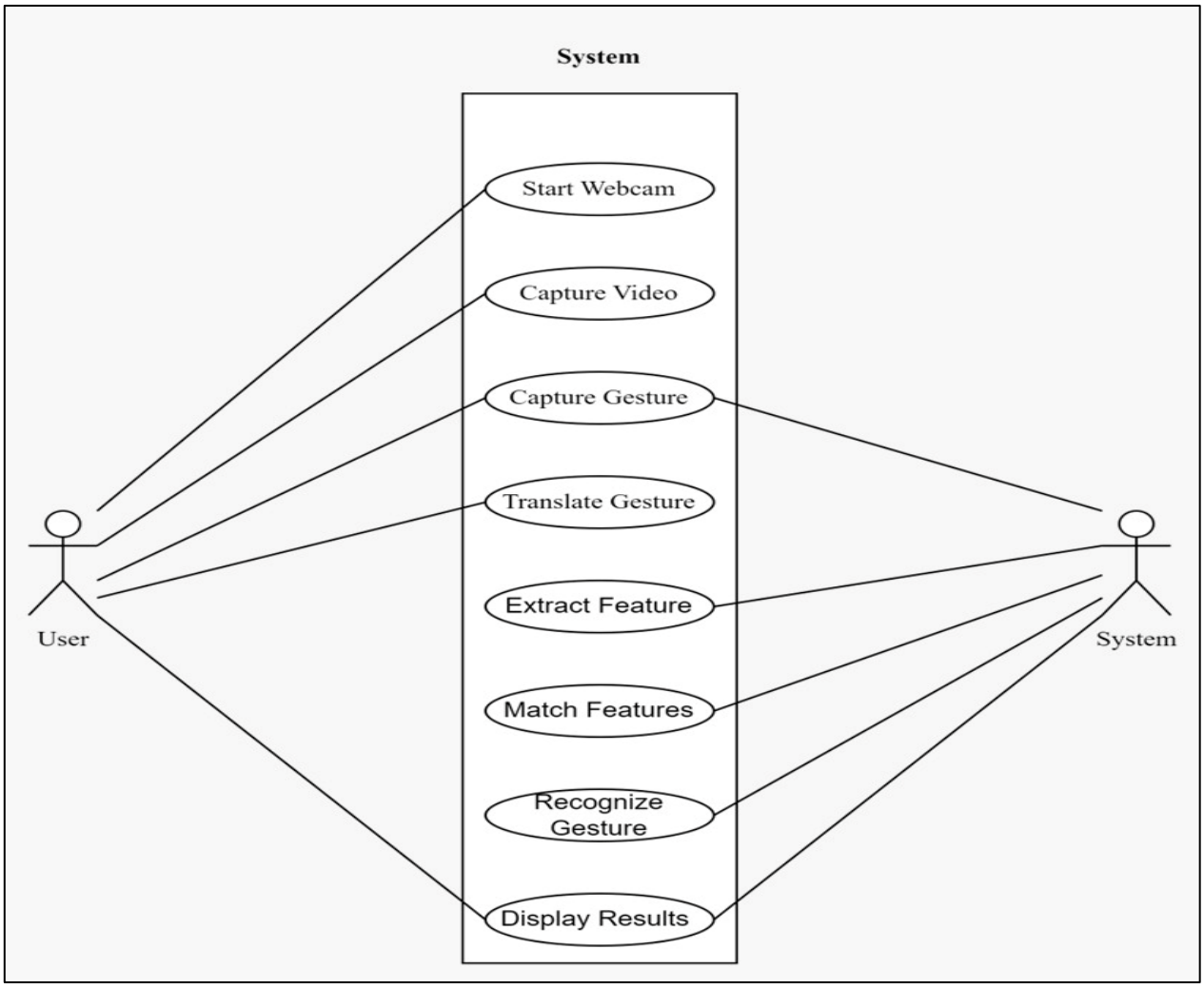


Figure 3.5.1 Use Case Diagram

3.5.2 Class diagram

Class diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different class involved. Figure class diagram. Shows use class diagram for the model, it shows the method that used in the proposed method.

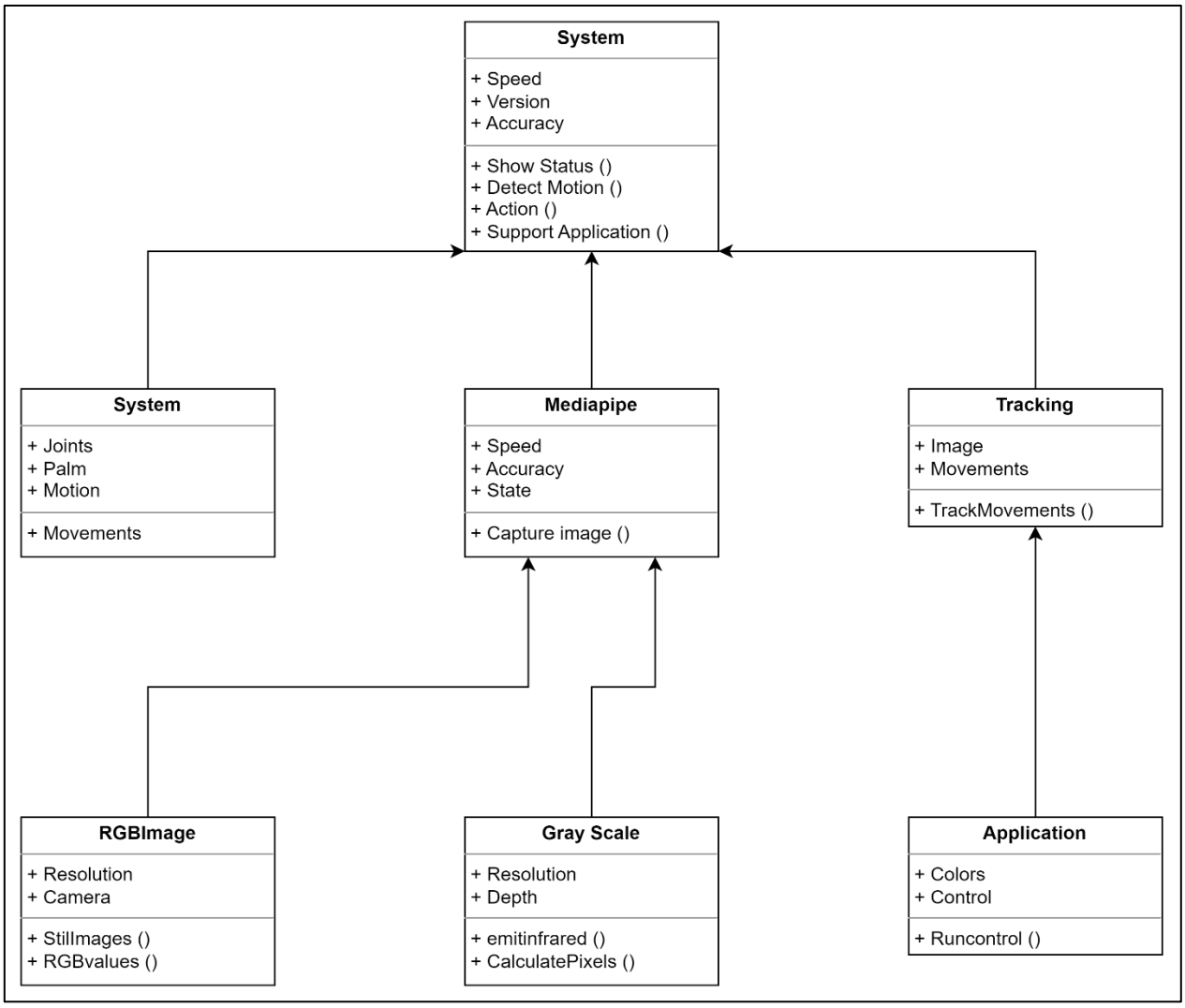


Figure 3.5.2 Class Diagram

A class diagram in UML is a static structure diagram which describes the structure of a system by showing the system’s classes, attributes, methods and the relationship between the object.

3.5.3 Activity Diagram

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

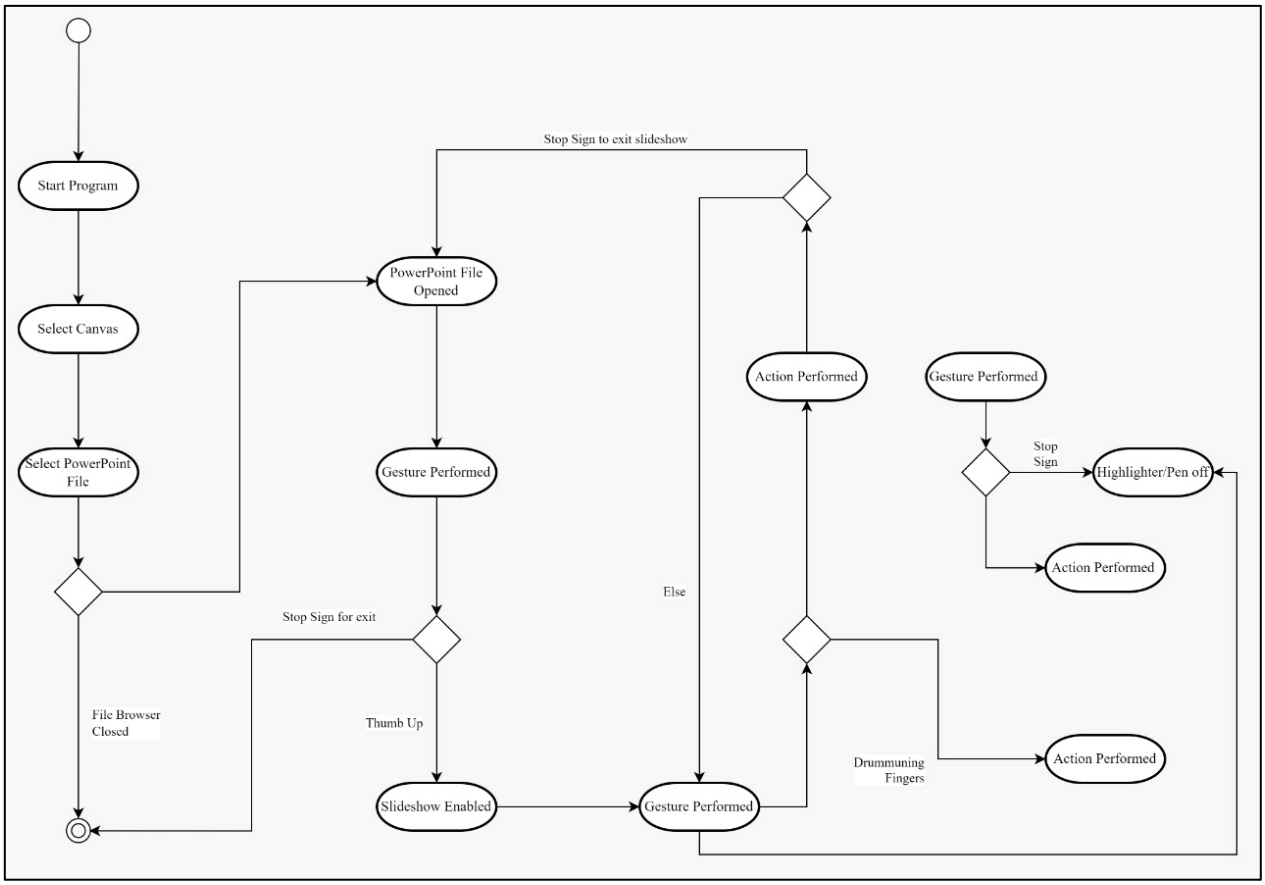


Figure 3.5.3 Activity Diagram

Chapter 4

Implementation plan for next semester

This chapter covers the activity table and Gantt chart for the proposed system. The main focus is to improve the accuracy of the model by tuning the hyper parameter and also using more dataset on it to enhance its accuracy. Connecting model functionality to tkinter GUI. After all this will test all the modules together and will fix identified bugs.

Sr. No.	Task Name	Start	Finish	Days
1	PPT control	27-11-2022	27-12-2022	30
2	Adding more colors to write	30-12-2022	06-01-2023	07
3	Adding more tools	08-01-2023	14-01-2023	06
4	Assigning tools using gestures.	23-01-2023	14-02-2023	22

Figure 4.1 Implementation Plan for Next Semester

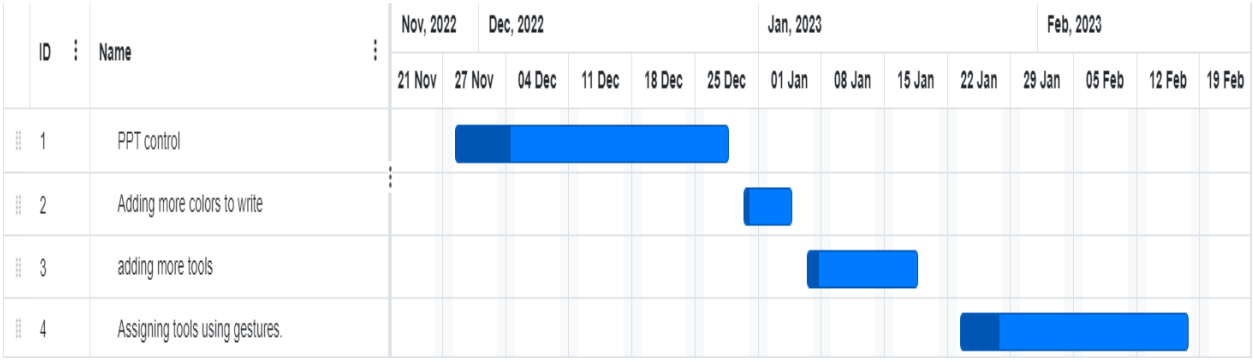


Figure 4.2 Gantt chart

Chapter 5

Partial Implementations and Results

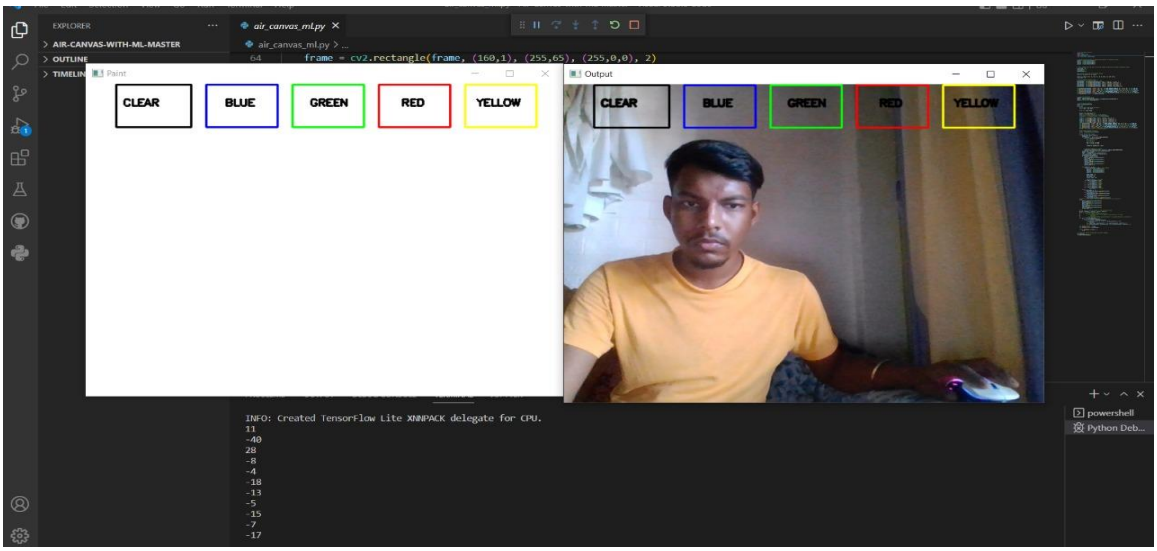


Figure 5.1 Preparing Canvas

In the Figure 5.1 Prepare the Canvas for Drawing.

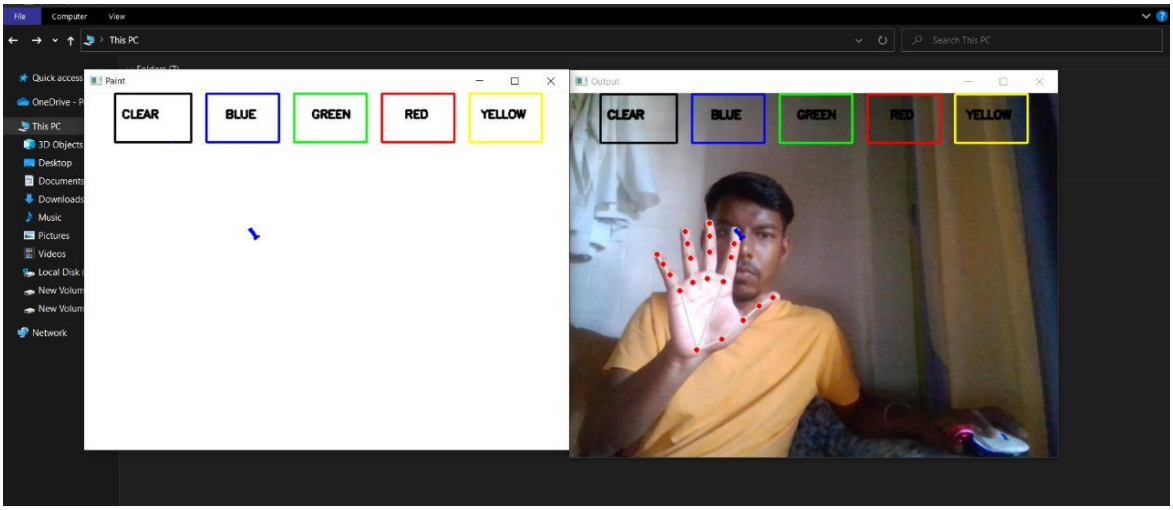


Figure 5.2 Landmark Detection

In the Figure 5.2 It detects the 21 landmarks of the hand given data by mediapipe module.

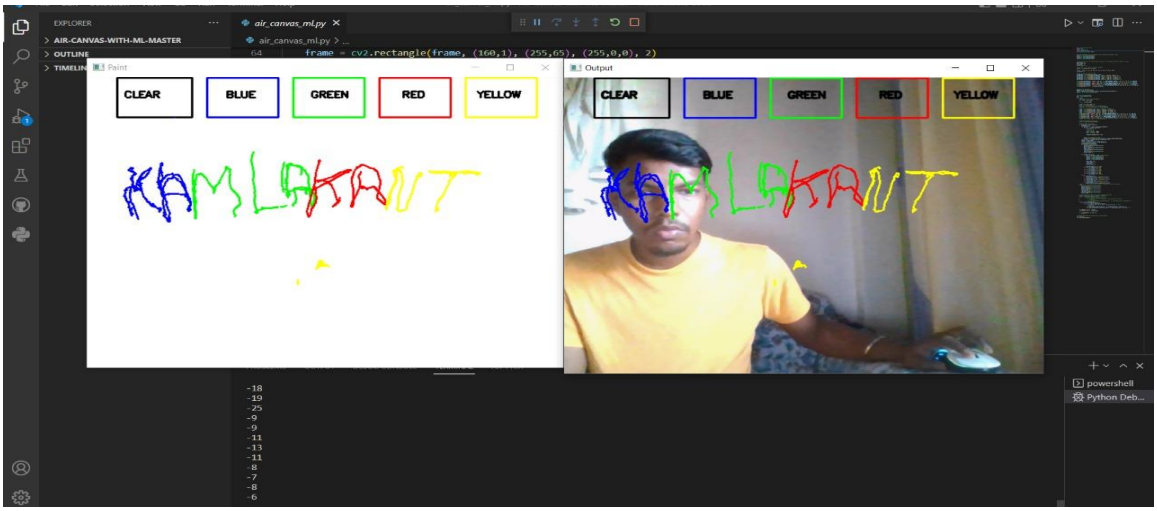
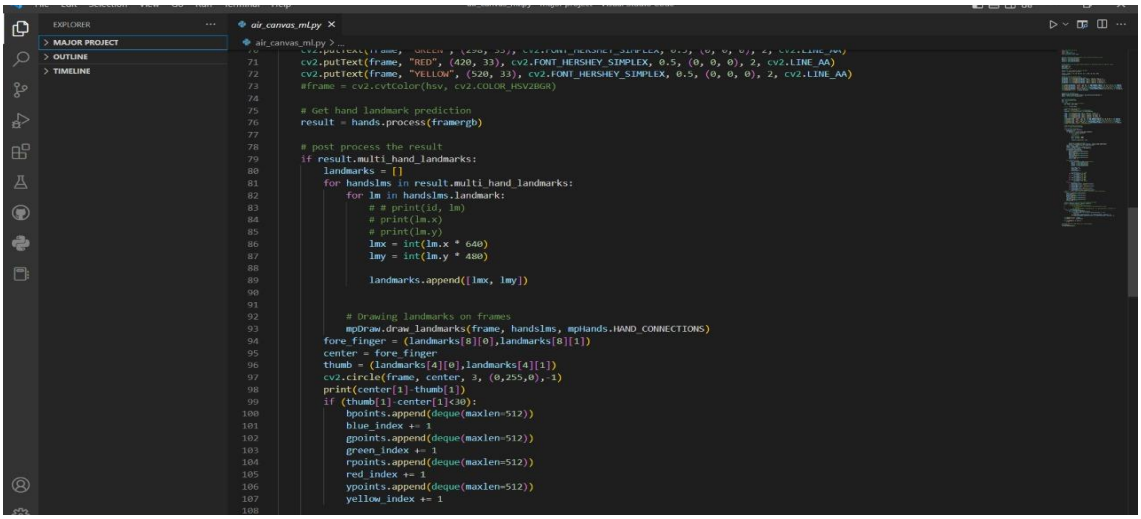


Figure 5.3 Performing Paint

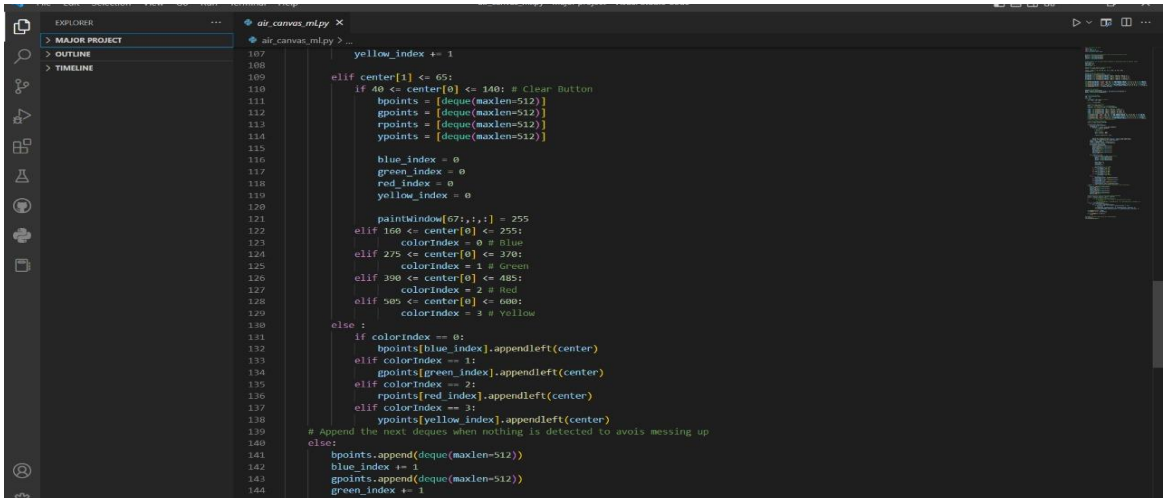
In the Figure 5.3 Drawing on Canvas using gestures and perform the operation like painting.



```
cv2.putText(frame, "RED", (420, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 0), 2, cv2.LINE_AA)
cv2.putText(frame, "YELLOW", (520, 33), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 0), 2, cv2.LINE_AA)
#frame = cv2.cvtColor(hsv, cv2.COLOR_HSV2BGR)

# Get hand landmark prediction
result = hands.process(framergb)

# post process the result
if result.multi_hand_landmarks:
    landmarks = []
    for handslms in result.multi_hand_landmarks:
        for lm in handslms.landmark:
            # # print(id, lm)
            # print(lm.x)
            # print(lm.y)
            lmx = int(lm.x * 640)
            lmy = int(lm.y * 480)

            landmarks.append([lmx, lmy])


        # Drawing landmarks on frames
        mpDraw.draw_landmarks(frame, handslms, mpHands.HAND_CONNECTIONS)
    fore_finger = (landmarks[8][0],landmarks[8][1])
    center = fore_finger
    thumb = (landmarks[4][0],landmarks[4][1])
    cv2.circle(frame, center, 3, (0,255,0),-1)
    print(center[1]-thumb[1])
    if (thumb[1]-center[1]<30):
        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
```

Figure 5.4 Landmark Prediction and Drawing Landmarks on frame

In the Figure 5.4 Getting hand landmark and processing the results or drawing landmarks on frame.



```
        yellow_index += 1

    elif center[1] <= 65:
        if 40 <= center[0] <= 140: # Clear Button
            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)
# Append the next deques when nothing is detected to avois messing up
else:
    bpoints.append(deque(maxlen=512))
    blue_index += 1
    gpoints.append(deque(maxlen=512))
    green_index += 1
```

Figure 5.5 Color Index

In the Figure 5.5 shows the color index is BGRY (Blue, Green, Red, Yellow) also shows the coordinates on paint window.

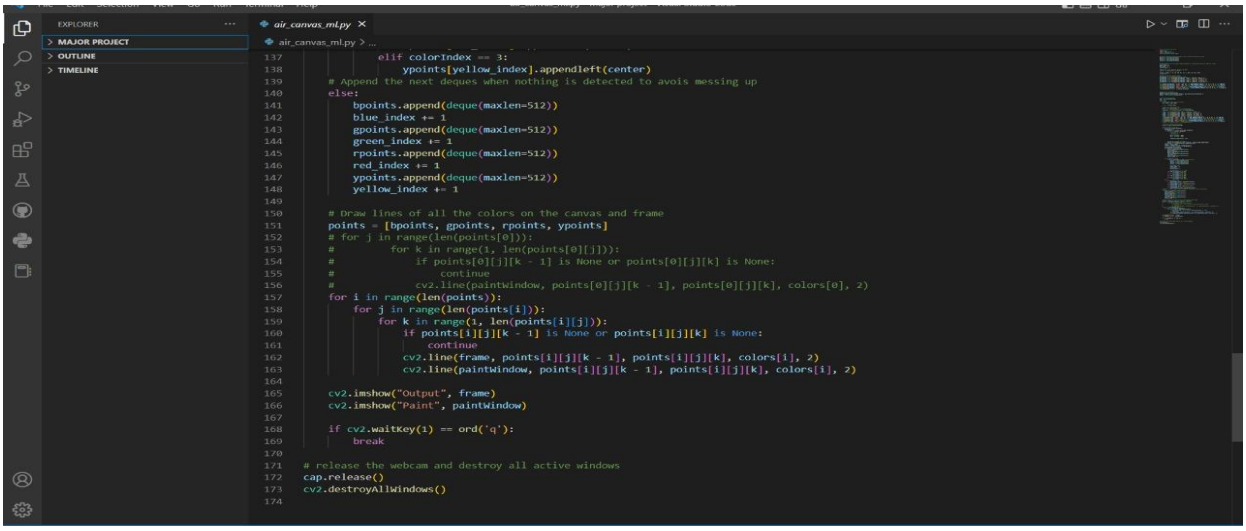


Figure 5.6 Deque

In the Figure 5.6 Appending the points to avoid complexity next deque when nothing is detected to avoid missing up. Draw lines of all the colors on the Canvas and frame. Release the webcam and destroy all active windows.

Chapter 6

Conclusion

This proposed system has the potential to challenge traditional writing methods. Eliminates the need to carry a cell phone in hand to take notes, to give an easy way on the go to do the same. It will again work towards a greater purpose in helping especially those who know them to communicate easily. Even adults who find it difficult to use the keyboard can easily use the program. Expanding functionality, this program can also be used to control IoT devices soon. Air painting can also be made happen. This program will be very good smart clothing software using which people can work better with the digital world. The unpopular reality of taxpayers we see can make the text come alive. Wind-writing programs should listen only to their master's control touch and should not be misled by people all around.

References:

[1] P.Vidhate, “Virtual Paint Application By Hand Gesture Recognition System” IEEE 2019

[2] P.Srungavarapu, P.maganti, S.Sakhamura, “Virtual Sketch using Open CV” IJITEE 2020

[3] T.saluke, “PowerPoint control using hand gesture recognition based on hog feature extraction and K-NN classification” IEEE 2019

[4] Z.Yuan, G.Jil, “Sketch recognition based intelligent whiteboard teaching system” ICCSSE 2008

[5] P.Kirchi, M.Cambek “Hand Gesture detection” IEEE 2019

[6] M.Idrees, M.Butt, A.Ahmad, H.Danish “Controlling PowerPoint using hand gesture in Python” ResearchGate 2021

[7] S.Saoji, N.Dua, A.Choudhary “Basic paint window Application via webcam using Open CV and NumPy in Python” ResearchGate 2021

[8] S.Kadam, A.Ghodke, S.Sadhukhan “Hand Gesture Recognition Software Based on Indian Sign Language.” IEEE 2019

[9] Y.Patil, M.Paun, D.Paun, K.Singh, V.Borate “Virtual Painting with Opencv Using Python.” IJSRST 2020

[10] Z.Hu, X.Zhu “Gesture detection from RGB hand image using modified convolutional neural network.” IEEE 2019

[11] I.Dhall, S.Vashisth, G.Aggarwal “Automated Hand Gesture Recognition using a Deep Convolutional Neural Network model.” IEEE 2020

[12] A.Ren, Y.Wang, X.Yang, M.Zhou “A Dynamic Continuous Hand Gesture Detection and Recognition Method with FMCW Radar.” IEEE 2020

[13] N.McHenry, L.Davis, I.Gomez III, N.Coute, N.Rochrs “Design of an AR Visor Display System for Extravehicular Activity Operations” IEEE 2020

[14] S.Bansode, S.Varkhad, S.Dhaigude, S.Waghmare, S.Suryawanshi “Computer Vision

Based Virtual Sketch Using Detection.” IJRASET 2022

[15] M.Rao, M.Niharika, J.Neha, P.Vidyashree, “Virtual Paint Application Using Hand Gestures” IRJET 2022

Acknowledgement

We would like to express a deep sense of gratitude towards our guide Prof. Monali Sankhe, Computer Engineering Department for her/his constant encouragement and valuable suggestions. The work that we are able to present is possible because of her/his timely guidance.

We would like to pay gratitude to the panel of examiners for the time, effort they put into evaluating our work and their valuable suggestions from time to time.

We would like to thank Project Head of the Computer Engineering Department, Prof. Janhavi Sangoi for her support and coordination.

We would like to thank Head of the Computer Engineering Department, Prof. Ashwini Save for her support and coordination.

We are also grateful to the teaching and non-teaching staff of the Computer Engineering Department who lend their helping hands in providing continuous support.

Kamlakant Bag

Siddharth Urankar

Ankita Yadav