;

AIR CANVAS USING HAND GESTURES

A PROJECT REPORT

Submitted by

R.ARUNASALAM (910719104002)

T.BHARATHI (910719104005)

S.BRINDHA (910719104006)

K.SIVARANJANI (910719104025)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE ENGINEERING

K.L.N COLLEGE OF INFORMATION TECHNOLOGY

SIVAGANGAI DISTRICT – 630612

ANNA UNIVERSITY::CHENNAI 600025

MAY 2023

ANNA UNIVERSITY CHENNAI::CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"AIR CANVAS USING HAND GESTURES "** is the bonafide work of "**R.ARUNASALAM (910719104002), T.BHARATHI(910719104005), S.BRINDHA(910719104006), K.SIVARANJANI(910719104025)**" who carried out the work under my Supervision.

SIGNATURE	**SIGNATURE**
Prof.N.M.Ram Kumar,M.A,M.Phil	**Prof.V.Merlifreeda,ME**
HEAD OF THE DEPARTMENT(i/c)	**SUPERVISOR**
PROFESSOR	**ASSISTANT PROFESSOR**
Department of CSE, K.L.N. College of Information Technology, Pottapalayam, Sivagangai-630612	Department of IT, K.L.N. College of Information Technology, Pottapalayam, Sivagangai-630612

Submitted for project Viva–Voce examination held on _______________

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

We, first of all would like to thank **GOD and our PARENTS** for giving us this wonderful opportunity to express our real talent.

Our sincere thanks to our honorable Principal and Professor & Head/Civil **Dr.K.R.YOGANATHAN** for giving us an opportunity to complete this project successfully. We would like to express our special thanks and gratitude to our guide

Merlifreeda.V,ME Asst.Prof Head of the Computer science Engineering Department, K.L.N. College of Information Technology who gave constant encouragement, guidance, ideas, assistance and support to successful completion of our project.

We would like to extend our special thanks to all our **TEACHING** and **NONTEACHING STAFF MEMBERS** for their constant encouragement and support which motivated us to complete our project on time.

Last but not least, our heartfelt thanks to **OUR FRIENDS** who helped and supportedus to complete our project work successfully.

ABSTRACT

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. It contributes immensely tothe advancement of an automation process and can improve the interface between man and machine in numerous applications. Several research works have been focusing on new techniques and methods that would reduce the processing time while providing higher recognition accuracy. Object tracking is considered as an important task within the field of Computer Vision. The invention of faster computers, availabilityof inexpensive and good quality video cameras and demands of automated video analysis has given popularity to object tracking techniques. Generally, video analysis procedure has three major steps: firstly, detecting of the object, secondly tracking its movement from frame to frame and lastly analysing the behaviour of that object. For object tracking, four different issues are taken into account; selection of suitable object representation, feature selection for tracking, object detection and object tracking. In real world, Object tracking algorithms are the primarily part of different applications such as: automatic surveillance, video indexing and vehicle navigation etc. method that reduces mobile and laptop usage by eliminating the need write to The project takes advantage of this gap and focuses on developing a motion-to-text converter that can potentially serve as software for intelligent wearable devices for writing from the air. This project is a reporter of occasional gestures. It will use computer vision to trace the path of the finger. The generated text can also be used for various purposes, such as sending messages, emails, etc. It will be a powerful means of communication for the deaf. It is an effective communication

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

Cv2	**-**	**Opencv**
HCI	**-**	**Human computer interaction**
RGB	**-**	**Red, Blue,Green**
HGR	**-**	**Hand gesture recognition**
AI	**-**	**Artifical intelligence**

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1

INTRODUCTION

1.1.GENERAL INTRODUCTION:

Air Canvas is a hands-free digital drawing canvas which utilizes camera, opencv and mediapipe 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 bemodified , also the pen color can be changed by hovering pointer over built-in buttons.The main objective of our project is you never thought ,raising your finger in the air can help us draw on a real picture. How wonderful that this aerial web works in computer vision projects. Computer projects helps us to draw on a screen easily by waving our fingers with a color indicated, Open CV helps in more advancement many ways of writing, such as using keyboards,touchscreen surfaces,digital pens,styluses,using electronic gloves, and more. But in this system, we use hand gesturerecognition with theuse of machine learning algorithm using python. Computer vision techniques are used todraw different shapes. This system uses python language to built the code. Camera and Mediapipe 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 areawhere we need to annotate or underline. In the era of digital world, traditional art of writing is being replaced by digital art. Digital art refers to forms of expression and transmission of art form with digital form. Relying on modern science and technology is the distinctive characteristics of the digital manifestation. Traditional art refers to the artform which is created before the digital art. From the recipient to analyse, it can simply be divided intovisual art, audio art, audio-visual art and audio-visual imaginary art, which includes literature driving force anyway. The same situation happens in art.

Artificial intelligence (AI)

Artificial intelligence (AI) is intelligence—perceiving, synthesizing, and inferring information—demonstrated by machines, as opposed to intelligence displayed by humans or by other animals. Example tasks in which this is done include speech recognition, computer vision, translation between (natural) languages, as well as other mappings of inputs.

As machines become increasingly capable, tasks considered to require "intelligence" are often removed from the definition of AI, a phenomenon known as the AI effect.[2] For instance, optical character recognition is frequently excluded from things considered to be AI, having become a routine technology.

Artificial intelligence was founded as an academic discipline in 1956, and in the years since it has experienced several waves of optimism, followed by disappointment and the loss of funding (known as an "AI winter"), followed by new approaches, success, and renewed funding. AI research has tried and discarded many different approaches, including simulating the brain, modeling human problem solving, formal logic, large databases of knowledge, and imitating animal behavior. In the first decades of the 21st century, highly mathematical and statistical machine learning has dominated the field, and this technique has proved highly successful, helping to solve many challenging problems throughout industry and academia.

The various sub-fields of AI research are centered around particular goals and the use of particular tools. The traditional goals of AI research include reasoning, knowledge representation, planning, learning, natural language processing, perception, and the ability to move and manipulate objects.[a] General intelligence (the ability to solve an arbitrary problem) is among the field's long-term goals. To solve these problems, AI researchers have adapted and integrated a wide range of problem-solving techniques, including search and mathematical optimization, formal logic, artificial neural networks, and methods based on statistics, probability, and economics.

AI also draws upon computer science, psychology, linguistics, philosophy, and many other fields.The field was founded on the assumption that human intelligence "can be so precisely described that a machine can be made to simulate it", This raised philosophical arguments about the mind and the ethical consequences of creating artificial beings endowed with human-like intelligence; these issues have previously been explored by myth, fiction,and philosophy . Computer scientists and philosophers have since suggested that AI may become an existential risk to humanity if its rational capacities are not steered towards beneficial goals. The term artificial intelligence has also been criticized for overhyping AI's true technological capabilities. Already, AI- and machine learning-enabled technologies are used in medicine, transportation, robotics, science, education, the military, surveillance, finance and its regulation, agriculture, entertainment, retail, customer service, and manufacturing.The future of artificial intelligence appears bright with continued advancements in technology. Investment in artificial intelligence reached $93.5 billion in 2021, according to Statista. The current trend for neural networks to grow larger will likely continue into the near future as more functionality

CHAPTER 2

LITERATURE SURVEY

2.1 "Basic Paint Window Application via Webcam Using OpenCV and Numpy in Python"

Author : Prof.S.U.Saoji,Akash Kumar Choudhary,Bharat Phogat,Nistha Dua

Methodology:

Many methods are used for hand gesture recognition in real-time. Prof.S.U.Saoji,Akash Kumar Choudhary,Bharat Phogat,Nistha Dua has proposed a new method of HCI (Human- Computer Interaction), that uses marker detection and tracking technique. Instead of having a mouse or touchpad, two colored markers are worn on the tips of the fingers to generate eight hand movements to provide instructions to a desktop or laptop computer with a consumer- grade camera .

2.2 "Deep Learning Based Real-Time Recognition of Dynamic Finger Gestures using a Data Glove," in IEEE Access, vol. 8, pp. 219923-219933,2020

Author: M. Lee and J. Bae

Methodology:

In the developed system uses a data glove-based approach to recognize real-time dynamic hand gestures. The data glove has ten soft sensors integrated in it that measure the joint angles of five fingers and are used to collect gesture data. Real-time gestures are recognized using techniques such as gesture spotting, gesture sequence simplification, and gesture recognition. Shomi Khan, M. Elieas Ali, Sree Sourav Das havedeveloped a system that uses a skin color detection algorithm to convert ASL (American Sign Language) into text from real-time

2.3 "Real Time Hand Gesture Recognition by Skin Color Detection for American Sign Language," 2019 4th International Conference on Electrical Information and Communication Technology (EICT),2019

Author: S. Khan, M. E. Ali, S. Das and M. M. Rahman

Methodology:

The skin color and hand shape differ from person to person, detecting the hand might be challenging. The technology uses two neural networks to overcome this. The SCD (Scalable color descriptor) neural network is the first. The picture pixels are fed intothe SCD neural network, which determines whether or not they are skin pixels. The second one is HGR (Hand gesture recognition) neural network to which the extracted features will be provided. The features will be extracted by two distinct algorithms namely Finding the fingertip and Pixel segmentation.

2.4 "An economical air writing system converting finger movements totext using web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), 2016,

Author: P. Ramasamy, G.Prabhu and R. Srinivasan

Methodology:

Have proposed a revolutionary technology in which the user can write the alphabet ortype whatever he or she wants by merely waving his or her finger over a colorful LED light source. Only the color of the LED is tracked to extract the movement of the finger sketching the alphabet. The color of the tracked object is changed to white, while the background is changed to black. The black and white frames are stitched together to createa single black and white image of the alphabet that the user wanted to draw.

2.5 "An Arduino Based Gesture Control System for Human-ComputerInterface," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA),2018,

Author: S.Belgamwar and S.Agrawal

Methodology:

Have developed a new HCI technique that incorporates a camera, an accelerometer,a pair of Arduino microcontrollers, and Ultrasonic Distance Sensors. The primary concept behind this interface is to capture motions using Ultrasonic Distance Sensors. The distance between the hand and the distance sensor is calculated to record the gestures.

2.6 "Skeleton-Based Dynamic Hand Gesture Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition

Author: Q. De Smedt, H. Wannous and J.Vandeborre

Methodology:

Used a skeleton-based model. They used the geometric shape of the hand toobtainan effective descriptor from the Intel Real-Sense depth camera's hand skeleton linkedjoints. The skeleton- based approach is better than the depth-based approach. Developed a virtual paint application that uses ball-tracking technology to track the hand gestures and write on the screen. They have used a glove with a ping pong ball attached to it as a contour a customizable airbrush model that uses the Leap Motion Controller, which can track hands, to create an immersive freehand painting experience.

CHAPTER 3

EXISTING SYSTEM

The existing system only works with your fingers and no highlighters, paints, or relatives. Identify and distinguish something like a finger from RGB image without depth sensor great challenge. Another problem is lack of top and movement under the pen. The system uses a one RGB camera that you can overwrite. From the depths discovery is impossible, jobs up and down of the pen cannot be traced. So, everything finger path is drawn, and the result the image will be abstract and unseen by model. Using real-time hand touch to change position the process from one region to another requires a lot of code care. In addition, the user should know many movement to control his plan adequately. The project focuses on solving some of the most important social issues problems. First of all, there are many hearing-impaired people problems in everyday life. While listening again listening is taken for granted, people don't have thiscommunicating with a disability using sign language. Most countries in the world cannot understand yours feelings and emotions outside the middle translator. Second, overuse Smartphones: causes accidents, stress, distractions, and other illnesses that people can nolonger toleratefind out. Although its portability and ease of use exist. very popular, its obstacles include life terrifying events. Waste paper is not uncommon. Many papers arewasted on writing, drawing, etc. A4 paper production requires about 5 liters of water. 93% sources come from trees, 50% of commercial waste is paper, 25% of landfills are paper, and the list goes to. Waste paper harms the environment through use of water andtrees and produce tons waste. On-air writing can solve these problems quickly. It will serve as a communication tool for the deaf. Your online text can be displayed in AR or translated into speech. One can write on the air quickly and continue to operate without much interruption. Also, writing in the air does not require paper. Everything is stored electronically.

3.1. ALGORITHM

3.1.1. FINGERTIP DETECTION ALGORITHM

Fingertip detection using tophat Fig. 4 demonstrates how the top-hat transform is usedin the proposed method. Fig. 2(a) shows an input hand image. Fig. 2(b) exhibits the result of opening, i.e., erosion followed by dilation. Fig. 2(c) shows the opening residue, that is, the difference between Fig. 2(a) and Fig. 2(b). We then find the center of gravity of the hand blob in Fig. 3(a). We use this point as a reference for locating five fingertips. The fingertip is defined as the furthest point in each finger segment from the reference point (Fig. 3(b)).

Fig.3.1.1.Raw results of the top-hat transform(a)hand(b)hand blob and(c)fingers.

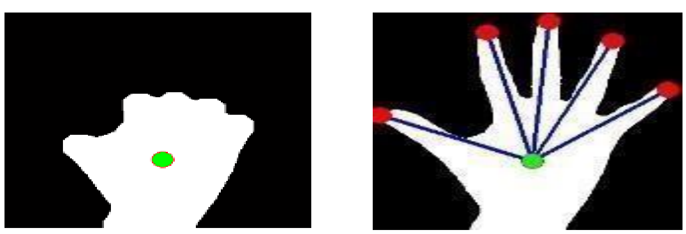

Fig.3.1.2.Final results of the top-hat transform(a)center of gravity and(b)fingertips

Table 1 shows a computation timetable. The convex hull algorithm is the fastestsince thealgorithm has a few steps that comprise finding the convex hull and subtracting it from the input to retrieve the convex deficiency. Finally, the points are marked on the hand contour where a transition is made in or out of a component of the convex deficiency. The KLT feature tracker is slightly slower than the convex hulleven when we slide the window along the contour of the segmented hand image. The SUSAN corner detector is the slowest since we have to count the number of pixels inside the circular mask for every spot when the circular mask is on the input image. The

proposed method is slightly slower than the convex hull and the KLT. It mostly spends time for finding the center of gravity anddistance measurement

Table1. Computation time.

	AVG (Sec)	SD (Sec)	MIN (Sec)	MAX (Sec)
Convex	0.1712	0.0089	0.1544	0.2006
KLT	0.2296	0.0359	0.1538	0.3579
SUSAN	0.6708	0.0866	0.4536	0.8221
Proposed	0.3763	0.0146	0.3518	0.4087

Table 3.1. Computation time.

The numerical results are also compared in sensitivity. In each approach, we record anumber of the input image which is successto detect fingertips on thumb, index, middle, ring, and little fingers. Since the proposed method has exactly 5 detected points, to make a fair comparison the existing methods are optimized by limiting the number of detected points to be 5, except for the convexhull algorithm.**Table 2**. Numerical results of the convex hull algorithm

Finger / Gesture	Thumb	Index	Middle	Ring	Little	Sensitivity	Precision
Opened	25	25	25	25	25	100%	100%
Semi-opened	25	25	25	25	25	100%	77.1%
Semi-closed	23	25	25	25	11	87.2%	68.1%
Closed	24	25	25	24	12	88%	60.9%

Table 3.1. Numerical results of the convex hull algorithm

Although the convex hull algorithm and the KLT tracker are fast, they commit numerous false detections. Simulation resultsreveal that the convex hull algorithm is effective only for opened hands where thefingers are spread widely. This is a serious limitation when the hand recognition is concerned. Meanwhile, the KLT feature tracker and the SUSAN corner detectorrespond not only to fingertips but also to the bases of the fingers. This means we need to introduce an extra process for distinguishing convex corners from concave corners. Simulation results also show that the KLT feature tracker and the SUSAN corner detector respond to the points where the curvature is locally maximal, rather than detecting the real fingertips. The proposedmethod, on the other hand, can successfully locate the fingertipsof the hands in various gestures, including a closed hand. Also, it is faster than the SUSAN corner detector

3.2. DISADVANTAGES:

The main disadvantage of interactive mode is that it can be difficult to automate tasks.

- Unfortunately, like other image processing problems, hand tracking and segmentation ina cluttered background is a critical problem in hand gesture recognition.
- In most occasions, the background is not simple. Also, the background may contain other body parts captured by the camera.

- The disadvantage is that some people may not pick up these signals, creating misunderstanding.

CHAPTER 4

PROPOSED SYSTEM

In the Proposed System we have utilized the advancements, for example, opencv and mediapipe. Hand motion acknowledgment can likewise be utilized for applications like modern robot control, communication through signing interpretation, in the restoration gadget for individuals with furthest point actual debilitations and so on. Hand motion acknowledgment finds applications in shifted spaces including virtual conditions, brilliant observation, gesture based communication interpretation, and so on. 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).

Step 1: Initialize Some Stuff

Firstly, we import the necessary libraries. Following are mentioned in the image:

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

Step 2: Setup the Paint Interface

Now we manually set the coordinates of each of the colour boxes on the frame. We use the OpenCV function cv2.rectangle() to draw the boxes.

Step 3: Start Reading the Video (Frame by Frame)

Now we use the OpenCV function cv2. VideoCapture() method to read a video, frame by frame (using a while loop), either from a video file or from a webcam in real-time. In this case, we pass 0 to the method to read from a webcam. We can just add the exact same paint interface for ease of usage.

Step 4: Find the Contour-Of-Interest

Once we start reading the webcam feed, we constantly look for a blue colour object in the frames with the help of cv2.inRange() method and use the blueUpper and blueLower variables initialized in Step 0. Once we find the contour, we do a series of image operations and make it smooth. These operations are specific type of morphing techniques as defined below.

4.1. ARCHITECTURE DIAGRAM

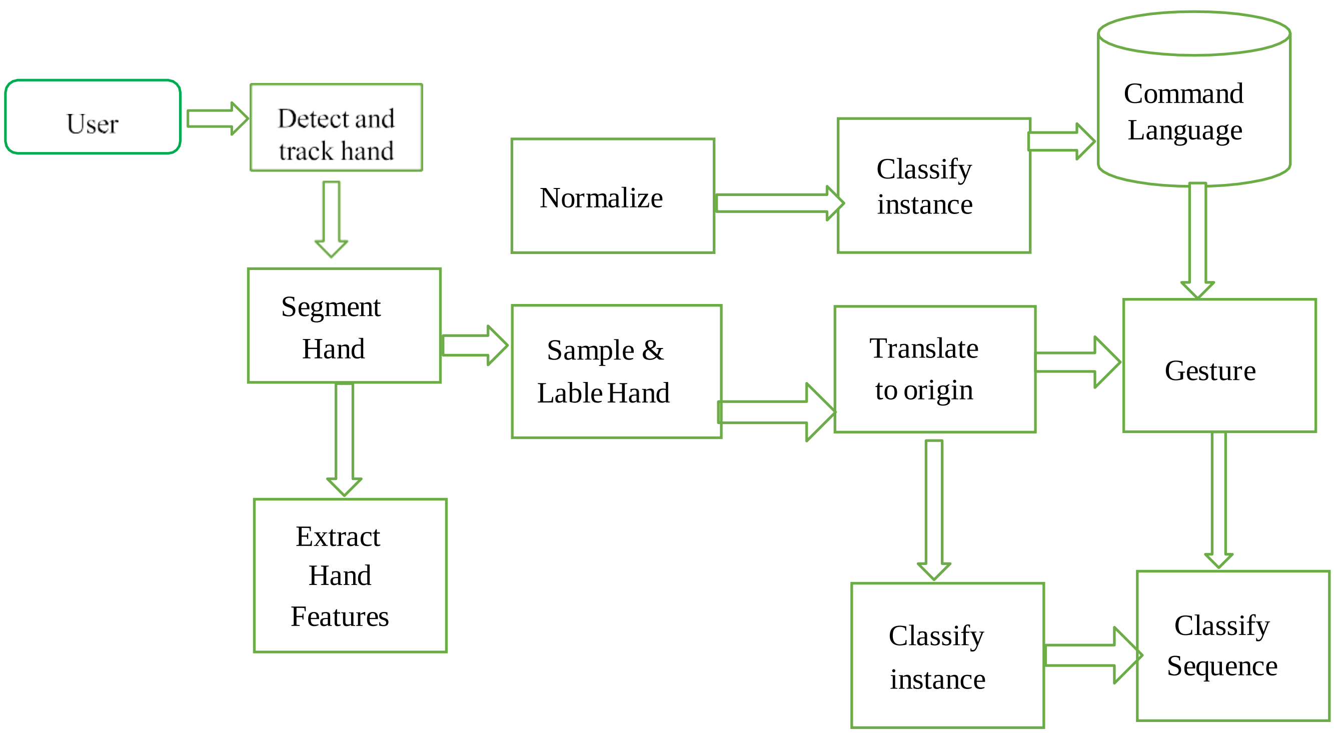

Fig.4.1.Architecture Diagram

4.2. FLOW DIAGRAM

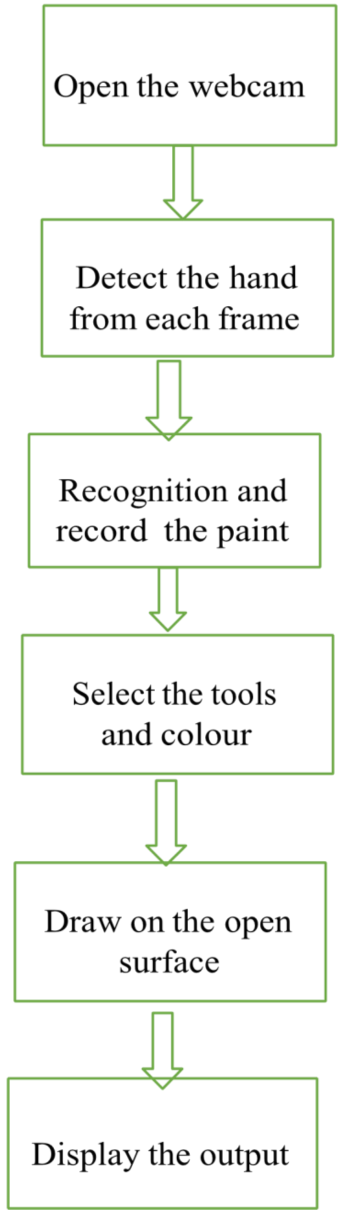

Fig.4.2.Flow Diagram

Based on the web camera frames that were captured, a virtual paint programme was offered. The webcamera sends the system the frames that it has received. Until the application is finished, the approachuses a web camera to collect each frame.

4.2.1. FEATURE EXTRACTION

Good segmentation process leads to perfect features extraction process and the latter paly an important role in a successful recognition process. The modules of OpenCV for image processing applications are given below:

1. CORE module contains image processing related function such as linear, non linear image filtering and geometrical image transformation etc.

2. IDEO module contains motion estimation and object tracking algorithms. Features vector of the segmented image can be extracted in different ways according to particular application. Various methods have been applied for representing the features can be extracted.

4.3. BLOCK DIAGRAM

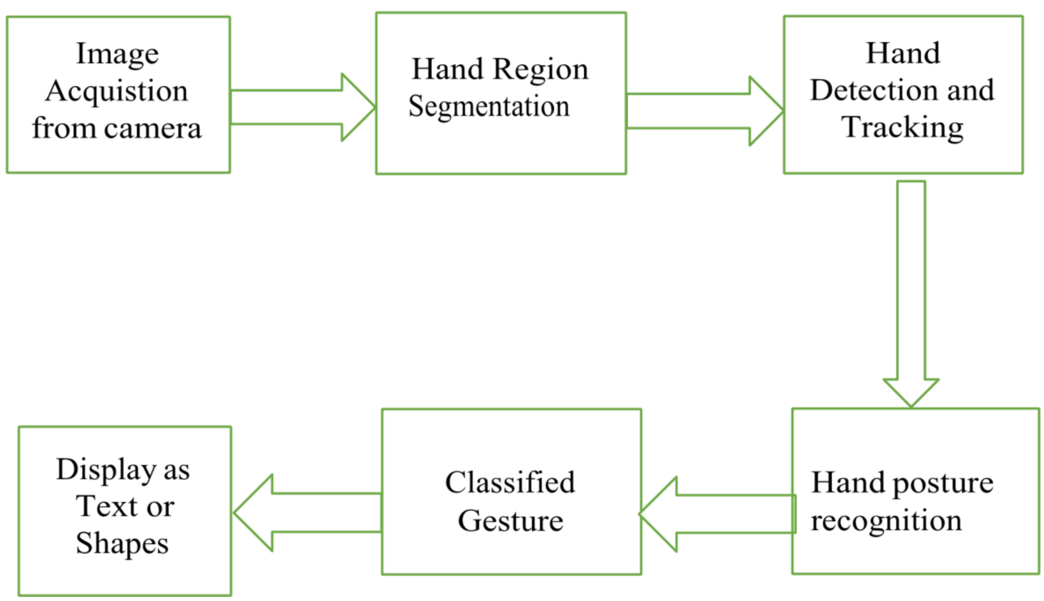

Fig.4.3.Block Diagram

First, Captured image from the webcam must be converted to the HRS color space to detect colored objects. The code below converts the incoming image to HRS space, which is great, and helps color space for color tracking.

4.4. IMPLEMENTATION

4.4.1. MODULES

- Data selection
- Preprocessing
- Data splitting
- ResultGeneration

4.4.2. MODULES DESCRIPTION

DATA SELECTION

- Given the real time webcam data.
- This application uses OpenCV library functions.
- The data selection is the process of recognizing of hand.

PREPROCESSING

- This paint like python application uses OpenCV library to track an object of interest and allows the user to draw by moving the object.
- Which makes it both awesome and challenging to draw simple things.

DATA SPLITTING

- In addition to the data required for training, test data are needed to evaluate the performance of the algorithm in order to see how well it works.
- In our process, we considered 70% of the input dataset to be the training data and the remaining 30% to be the testing data.
- Typically, when you separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing.

RESULT GENERATION

- The Final Result will get generated based on the overall classification and prediction.
- The performance of this proposed approach is evaluated using some measures like accuracy , precision and recall.

4.5. ALGORITHM

EXPLANATION HAND

REGION SEGMENTATION

Once we have accurately captured the hand using the above technique, the segmentation of the hand area is done using a two-step approach, viz. The skin segmentation and the subtraction of the background and the final binary image of the hand are obtained as an aggregation of the two. The proposed algorithm works well in real-time and provides relatively accurate segmentation results. Although skin colors vary greatly from breed to breed, it has been observed that skin color has a small area between different skin types, while skin luminosity differs significantly

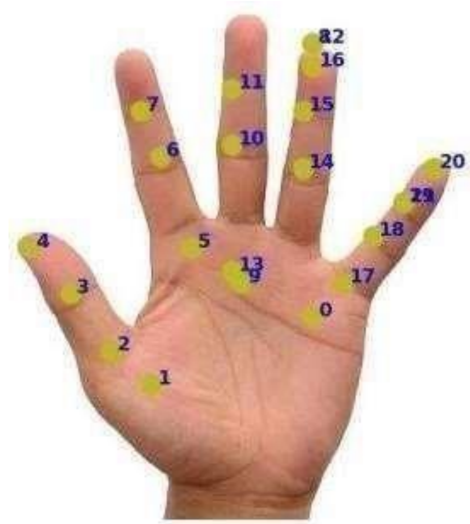

Fig.4.6.1. Hand Region Segmentation

Background subtraction: Since accurate hand detection with the Faster R-CNN handheld detector followed by filtering of skin color at the boundary of the candidate's hand provides a reasonably good segmentation result, the subtraction step of background is only used to remove skin-colored objects (not part of the hand) that are in the bounding box of the recognized hand may be present.

HAND DETECTION AND RECOGNITION

The mechanism between the brush in pace with the fingers movement. Generating multidimensional control to control properties. The painting model relates to the interaction mechanism between the brush and the air canvas. Since determining the exact center of gravity of the hand is critical in the following steps, the system uses two algorithms to determine the initial estimates of the center of gravity, and the final center of gravity is calculated as the average of the two. The distance transformation method is used to obtain the first estimate of the center of gravity (xc1, yc1). In the distance transform image, each pixel is represented by its distance from the next edge pixel. Euclidean distance was used to measure the distance between a pixel and its closest edge pixel. Therefore, the pixel with the highest intensity in the distance transform image is taken as the center of gravity.

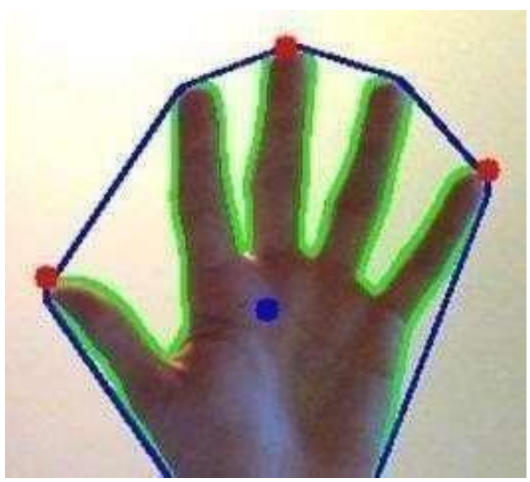

Fig.4.6.2.Hand Centroid Localization

HAND GESTURE CLASSIFIED

To archieve the tight combination of the operational plane and the display plane , we need to sense the hands movement relative to the screen . Hand detection and tracking on consecutive images is an essential part of fingertip detection and tracking.

Experiments show that using the Faster R- CNN handheld detector for each frame is computationally intensive and produces frame rates well below real-time performance. Therefore, the KCF tracking algorithm is used to track the area of the captured hand

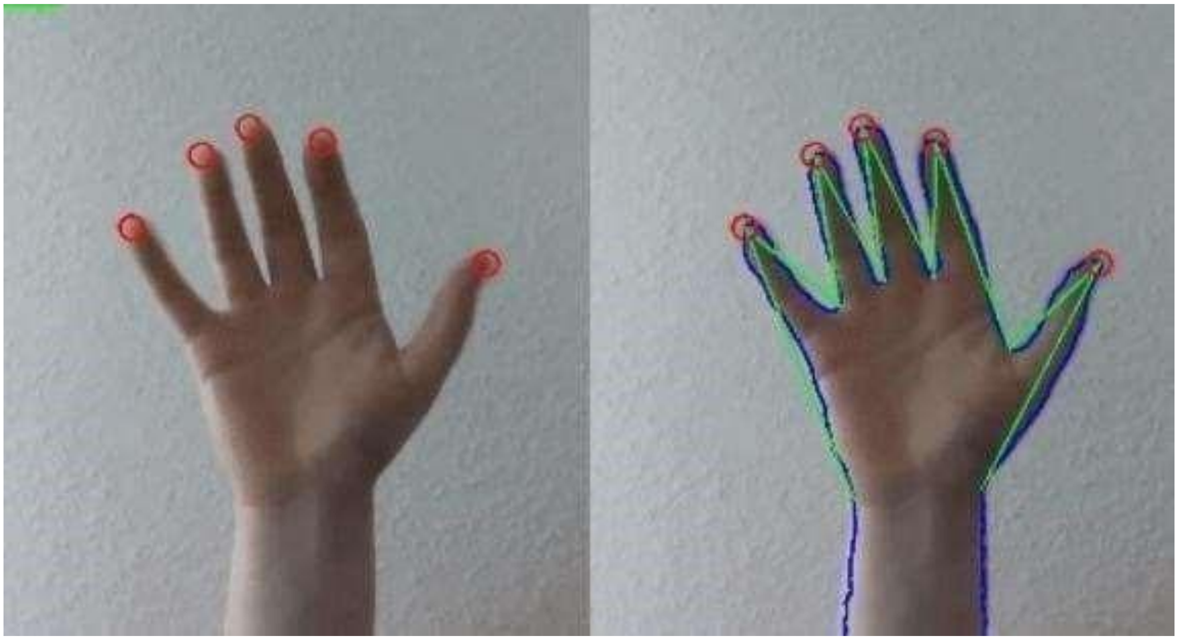

Fig.4.6.3.FingertipTracking

COLOUR TRACKING OF OBJECT AT FINGERPRINT

Incoming webcam images must be converted to HSV color space to recognize hand- colored objects.

z

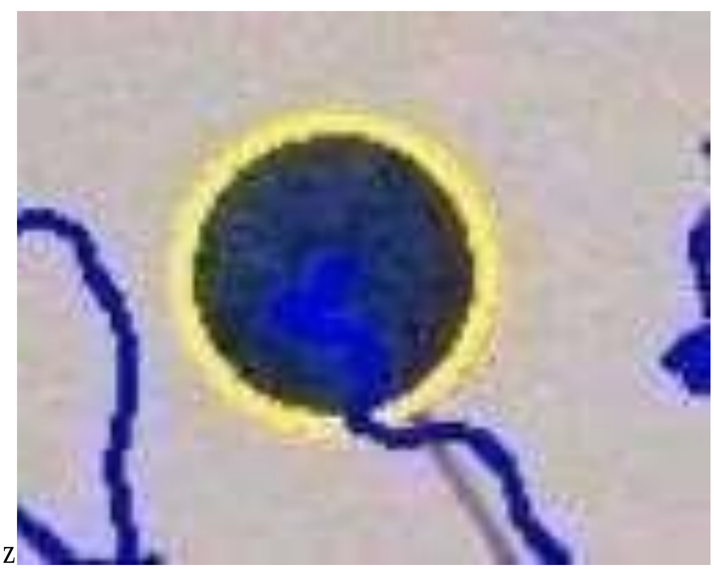

Fig.4.6.4. Colour Tracking of Object at Fingertip

The algorithm code converts the incoming image into HSV space, which is a very suitable color space and perfect for color tracking.

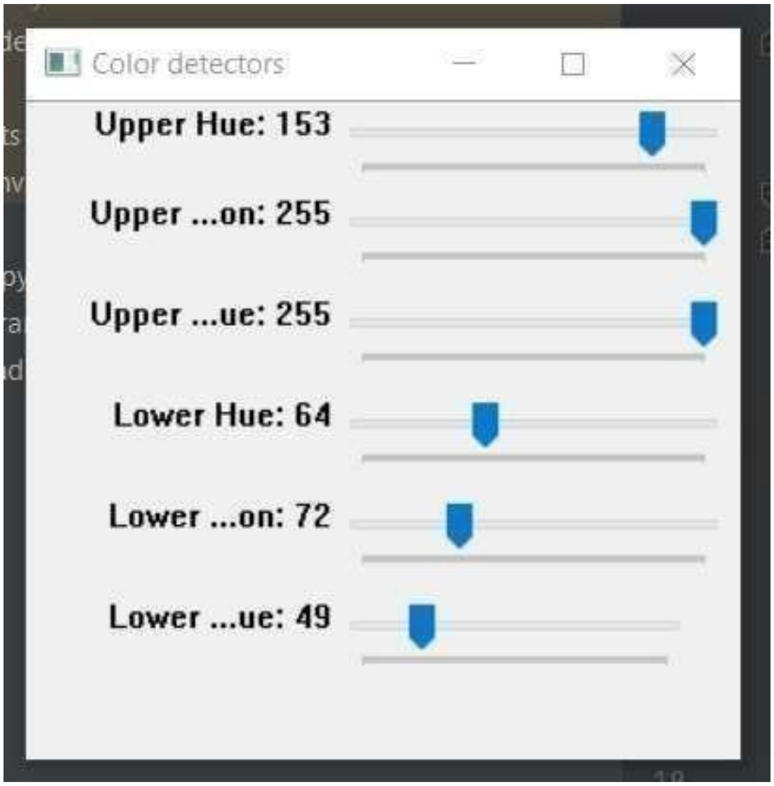

Fig.4.6.5.ColourTrackbar

The tracking bars have organized the HSV values into the required color range of the colored object you placed on your finger. Contour Detection of the Mask of Colour After detecting the Mask in Air Canvas, it is now time to locate its center position for drawing the Line. The system will perform some morphological operations on the Mask to make it free of impurities and detect contour easily.

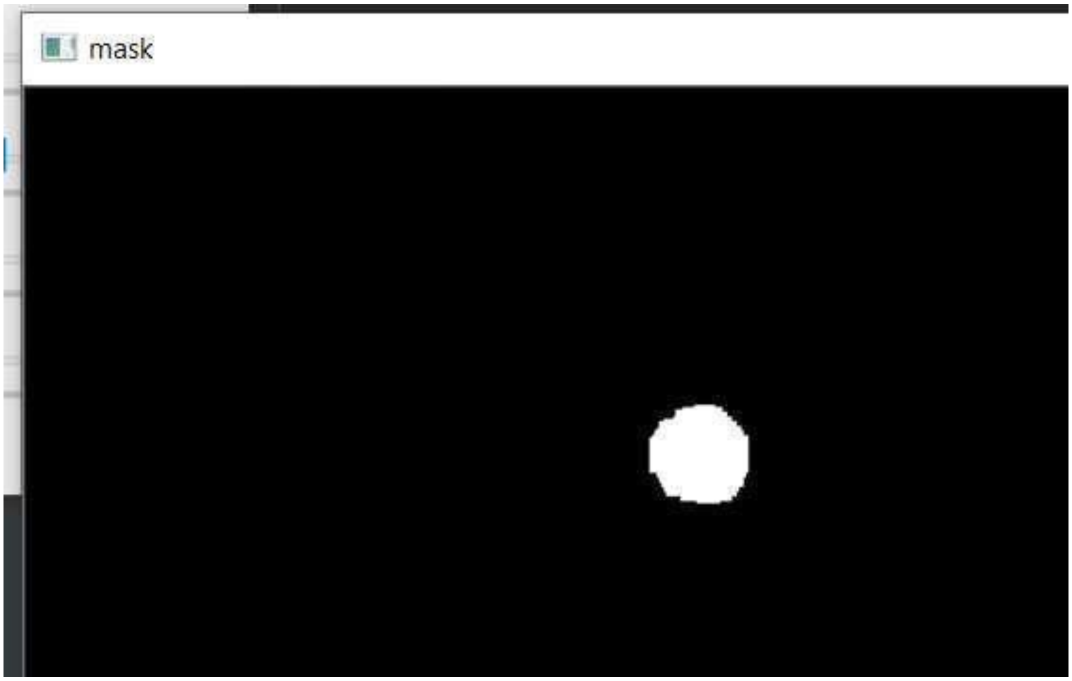

Fig.4.6.6.Mask Detection

DRAWING THE LINE USING THE POSITION OF THE CONTOUR

The actual logic behind this computer vision project is to create a Python deque (a data structure). 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. Now use the outline position to decide whether to click a button or draw on the provided sheet. Some of the buttons are at the top of the canvas. When the pointer enters this area, it is activated according to the method present in this area.

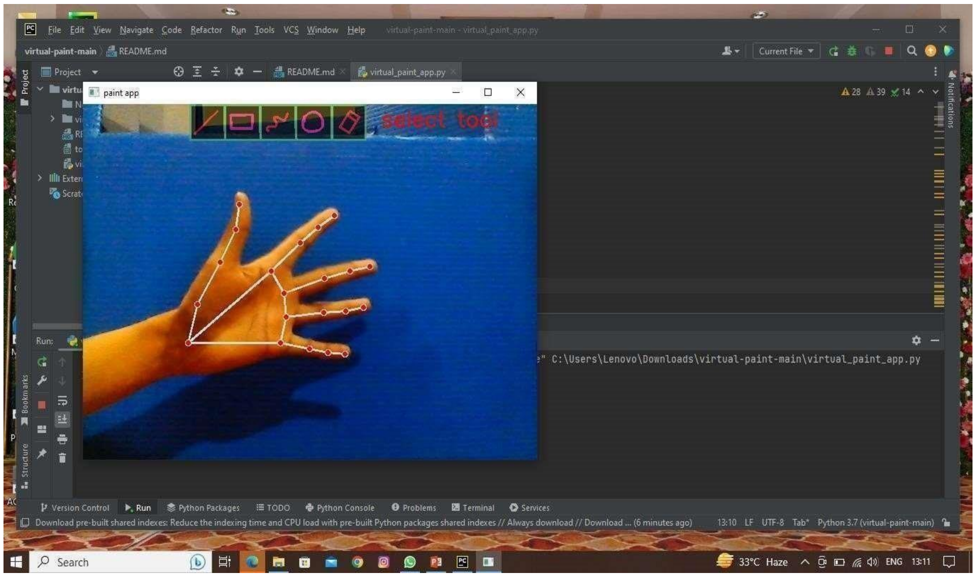

Fig.4.6.7.Air canvas using hand gesture

Object position: Image extracted from the video sequence. Extract the color image from the reference: This proposed method tracks the index finger movement, colored blue. We don't have a reference image, so any previous image is a reference to the next. Now take the difference in the images and extract the color and movement of the object. Edge Enhancement (EE) - Edge enhancement technique makes the object location algorithm robust against noise, different lighting conditions, darkening, and fading of objects, even in low contrast images.

DISPLAY THE PAINTING

The proposed airbrush models both the user brush and the brush canvas interactions. The result shows that this technology enables immersive freehand painting on a common screen with air canvas .

ADVANTAGES:

It forces you to work quickly to develop an eye for what the important elements are and capture those in quick gestures

- A whole painting studio on the palm of your hand.
- It trains your muscles to respond in a certain way. The more often you draw circles.
- It's not necessarily any different than any other painting media other that there no brushes and paints.

Artificial intelligence virtual painter is used to learn easily for kids in future you can Select the different colors and the eraser to erase the drawing.

The system has the potential to challenge tradition writing/teaching methods. The ultimate goal is to create a computer vision machine learning application that promotes human computer interaction also named Man -Machine interaction refers to therelation between the human and the computer or more precisely the machine.

CHAPTER 5

RESULTS AND DISCUSSION

5.1. SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- RAM : 4GB
- Storage : 500GB
- CPU : 2 GHz or faster
- Architecture : 32-bit or 64-bit

SOFTWARE REQUIREMENTS:

- Operating system : windows11
- Programming language : : Python
- Tools : Pycharm
- Python version : 3.7

5.2. SOFTWARE DESCRIPTION:

PYTHON

Python is one of those rare languages which can claim to be both simple and powerful. You will find yourself pleasantly surprised to see how easy it is to concentrate on the solution to the problem rather than the syntax and structure of the language you are programming in. The official introduction to Python is Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. I will discuss most of these features in more detail in the next section.

FEATURES OF PYTHON

Simple,

Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English, although very strict English! This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the language itself. **Easy to Learn**

As you will see, Python is extremely easy to get started with. Python has an extraordinarily simple syntax, as already mentioned.

Free and Open Source ,

Python is an example of a FLOSS (Free/Libré and Open Source Software). In simple terms, you can freely distribute copies of this software, read its source code, make changes to it, and use pieces of it in new free programs. FLOSS is based on the concept of a community which shares knowledge. This is one of the reasons why Python is so good - it has been created and is constantly improved by a community who just want to see a better Python.

High-level Language,

When you write programs in Python, you never need to bother about the lowlevel details such as managing the memory used by your program, etc. **Portable**

Due to its open-source nature, Python has been ported to (i.e. changed to make it work on) many platforms. All your Python programs can work on any of these platforms without requiring any changes at all if you are careful enough to avoid any system-dependent features.

You can use Python on GNU/Linux, Windows, FreeBSD, Macintosh, Solaris, OS/2, Amiga, AROS, AS/400, BeOS, OS/390, z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE and PocketPC!

You can even use a platform like Kivy to create games for your computer and for iPhone, iPad, and Android.

Interpreted

This requires a bit of explanation.

A program written in a compiled language like C or C++ is converted from the source language i.e. C or C++ into a language that is spoken by your computer (binarycode i.e. 0s and 1s) using a compiler with various flags and options. When you run the program, the linker/loader software copies the program from hard disk to memory and starts running it.

Python, on the other hand, does not need compilation to binary. You just run the program directly from the source code. Internally, Python converts the source code into an intermediate form called bytecodes and then translates this into the native language of your computer and then runs it. All this, actually, makes using Python much easier since you don't have to worry about compiling the program, making sure that the proper libraries are linked and loaded, etc. This also makes your Python programs much more portable, since you can just copy your Python program onto another computer and it just works!

Object Oriented

Python supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions which are nothing but reusable pieces of programs. In object-oriented languages, the program is built around objects which combine data and functionality. Python has a very powerful but simplistic way of doing OOP, especially when compared to big languages like C++ or Java.

Extensible

If you need a critical piece of code to run very fast or want to have some piece of algorithm not to be open, you can code that part of your program in C or C++ and then use it from your Python program.

Embeddable

You can embed Python within your C/C++ programs to give scripting capabilities for your program's users.

Extensive Libraries

The Python Standard Library is huge indeed. It can help you do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, FTP, email, XML, XML-RPC, HTML, WAV files, cryptography, GUI (graphical user interfaces), and other system-dependent stuff. Remember, all this is always available wherever Python is installed. This is called the Batteries Included philosophy of Python. Besides the standard library, there are various other high-quality libraries which you can find at the Python Package Index.

5.3. RESULT SCREENSHOT

SELECT TOOL

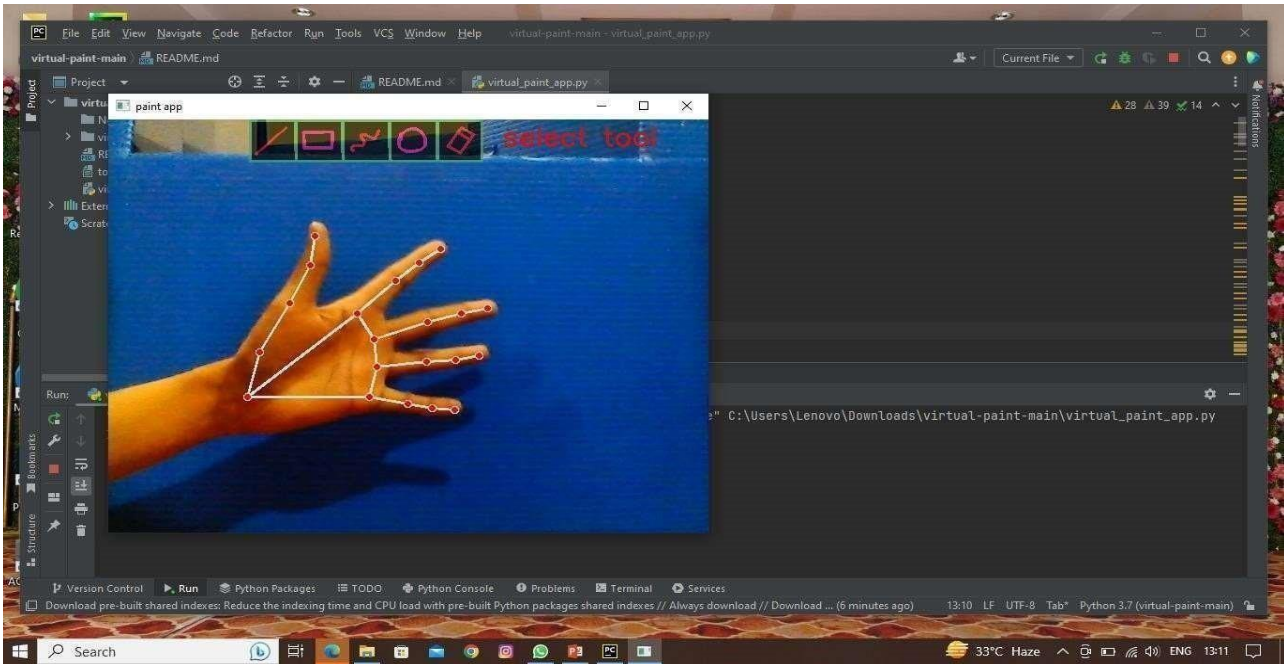

Fig.5.3.1.Select tool

RECTANGLE

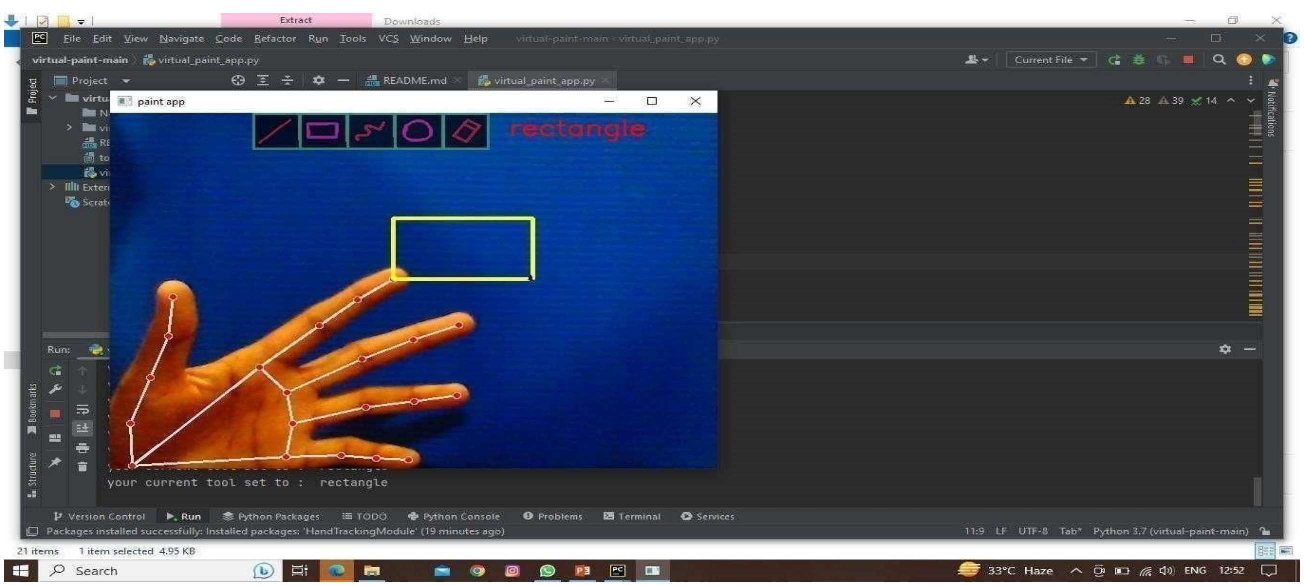

Fig.5.3.2.Rectangl

CIRCLE

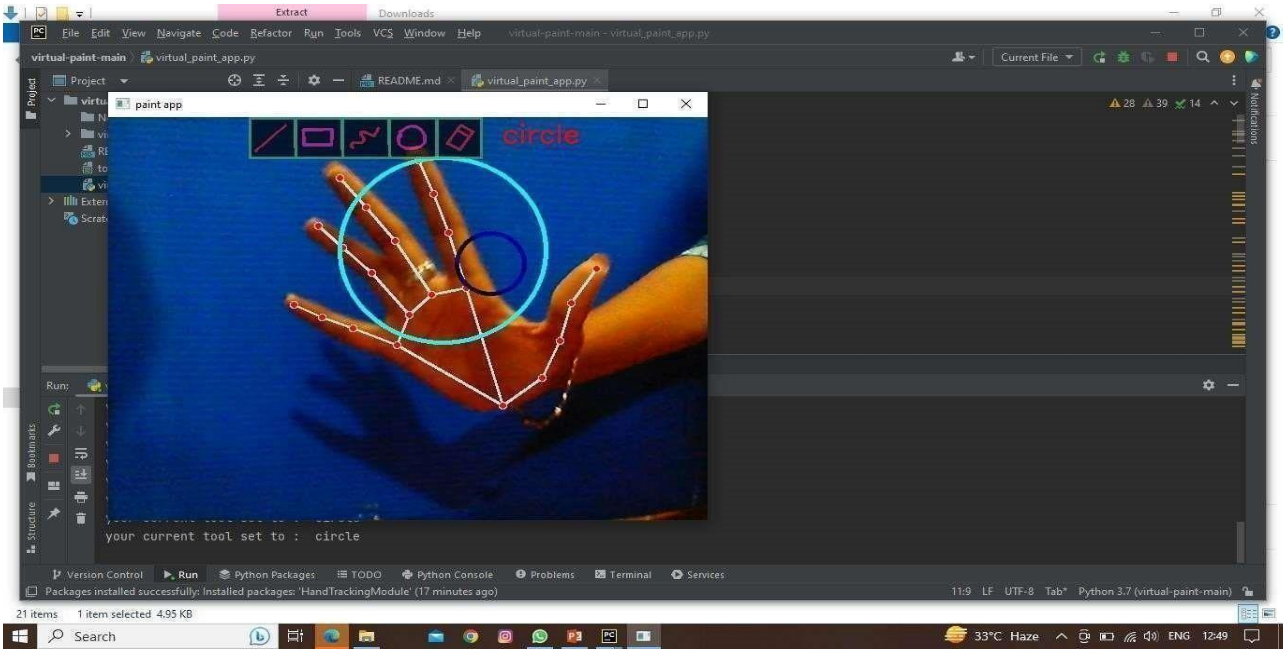

Fig.5.3.3. Circle

LINE

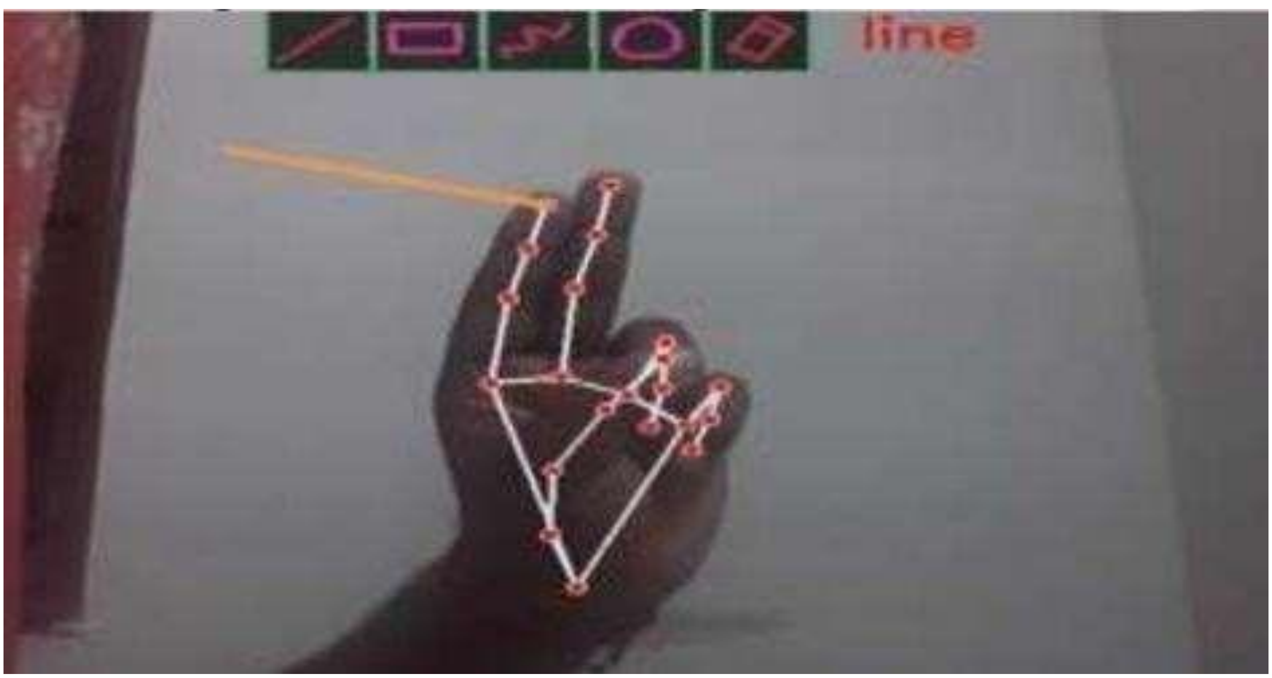

Fig.5.3.4.Line

DRAW

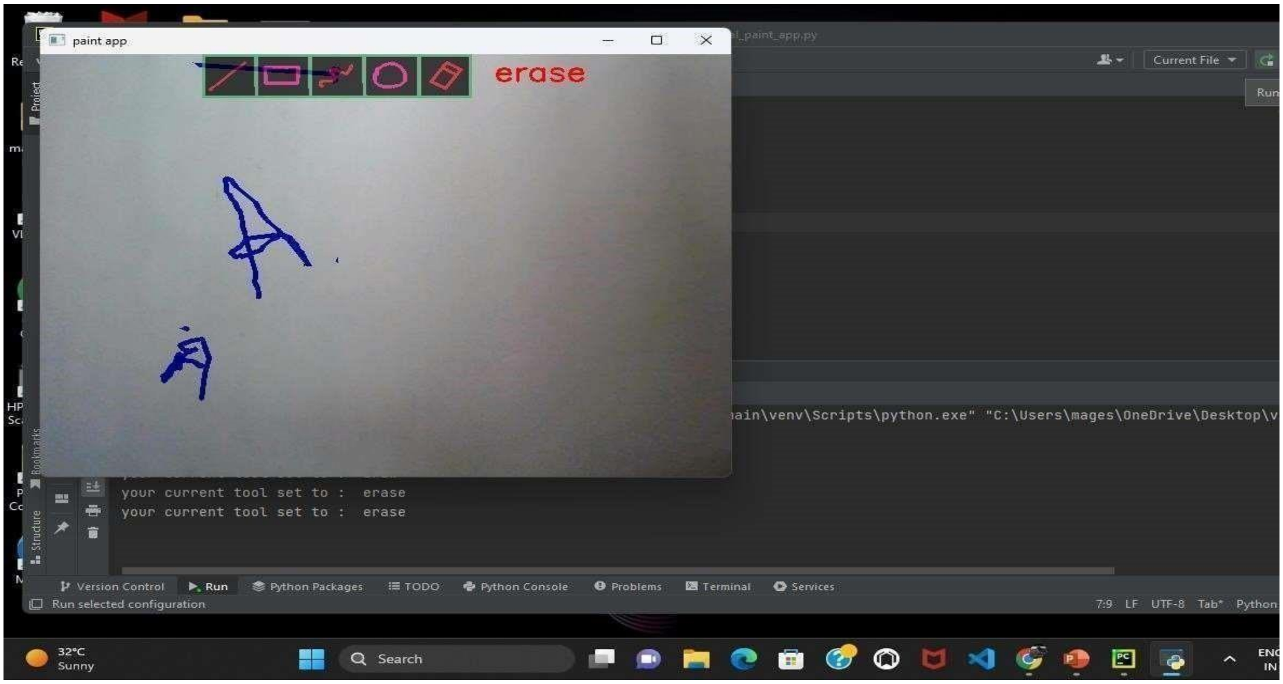

Fig.5.3.5.Draw

CHAPTER 6

SAMPLE CODING

```
import mediapipe as mp
import cv2 import
numpy as np import
time
#contants ml = 150 max_x,
max_y = 250+ml, 50
curr_tool = "select tool"
time_init = True rad = 40
var_inits = False thick = 4
prevx, prevy = 0,0

 #get tools
 function def
 getTool(x): if x <
 50 + ml: return
 "line"

  elif x<100 + ml:
  return "rectangle"
  elif x < 150 + ml:
  return"draw" elif
  x<200 + ml:
  return "circle"
```

```
else:
 return "erase"


 def index_raised(yi, y9): if
 (y9 - yi) > 40:
 return True return
 False

 hands = mp.solutions.hands hand_landmark =
 hands.Hands(min_detection_confidence=0.6,
 min_tracking_confidence=0.6, max_num_hands=1) draw =
 mp.solutions.drawing_utils
 #        drawing        tools   tools   =
 cv2.imread("tools.png") tools = tools.astype('uint8')

 mask = np.ones((480, 640))*255 mask
     =  mask.astype('uint8')'''
 tools = np.zeros((max_y+5, max_x+5, 3), dtype="uint8") cv2.rectangle(tools, (0,0), (max_x,
 max_y), (0,0,255), 2)

cv2.line(tools, (50,0), (50,50), (0,0,255), 2)
    cv2.line(tools, (100,0), (100,50), (0,0,255), 2)
     cv2.line(tools, (150,0), (150,50), (0,0,255), 2)
      cv2.line(tools, (200,0), (200,50), (0,0,255), 2)

      '''

 Cap = cv2.VideoCapture(0) while

 True: _, frm = cap.read()

 frm = cv2.flip(frm, 1)
```

```
rgb = cv2.cvtColor(frm, cv2.COLOR_BGR2RGB)

op = hand_landmark.process(rgb)
if op.multi_hand_landmarks: for i in op.multi_hand_landmarks:
draw.draw_landmarks(frm, i, hands.HAND_CONNECTIONS) x, y
= int(i.landmark[8].x*640), int(i.landmark[8].y*480)

if x < max_x and y < max_y and x > ml: if
time_init:ctime = time.time()
time_init = False
ptime = time.time()
cv2.circle(frm, (x, y),
rad,
(0,255,255), 2) rad
-= 1
if (ptime - ctime) > 0.8: curr_tool
=  getTool(x)     print("your current tool set to : ",
curr_tool) time_init
= True
rad = 40
else: time_init =
True rad = 40

if curr_tool == "draw":
xi, yi = int(i.landmark[12].x*640), int(i.landmark[12].y*480) y9
= int(i.landmark[9].y*480)     if
index_raised(yi, y9): cv2.line(mask,
(prevx,prevy), (x, y), 0, thick) prevx,
prevy = x, y

else: prevx = x prevy = y
elifcurr_tool == "line":
xi, yi = int(i.landmark[12].x*640), int(i.landmark[12].y*480) y9
```

```
= int(i.landmark[9].y*480)

if    index_raised(yi, y9):


if not(var_inits): xii, yii
 =        x,        y
 var_inits = True

cv2.line(frm, (xii, yii), (x, y), (50,152,255), thick)


 else: if var_inits: cv2.line(mask, (xii, yii), (x, y),0,
 thick) var_inits
      =   False   elif   curr_tool   ==   "rectangle":   xi,   yi   =
int(i.landmark[12].x*640),    int(i.landmark[12].y*480)    y9    =
int(i.landmark[9].y*480)

if index_raised(yi, y9): if
 not(var_inits): xii, yii =
 x, y var_inits = True

cv2.rectangle(frm, (xii, yii), (x, y), (0,255,255), thick)
 else: if var_inits: cv2.rectangle(mask, (xii, yii), (x, y),
 0, thick) var_inits
 = False

elif curr_tool == "circle":
 xi, yi = int(i.landmark[12].x*640), int(i.landmark[12].y*480)


y9 = int(i.landmark[9].y*480)


 if index_raised(yi, y9):
 if   not(var_inits):  xii,
```

```
yii = x, y var_inits =
True

cv2.circle(frm, (xii, yii), int(((xii-x)**2 + (yii-y)**2)**0.5), (255,255,0),

thick)else: if
var_inits:
cv2.circle(mask, (xii, yii), int(((xii-x)**2 + (yii-y)**2)**0.5), (0,255,0), thick) var_inits
= False

elif curr_tool == "erase":
xi, yi = int(i.landmark[12].x*640), int(i.landmark[12].y*480) y9
= int(i.landmark[9].y*480)

if index_raised(yi, y9):
cv2.circle(frm, (x, y), 30, (0,0,0), -1) cv2.circle(mask,
(x, y), 30, 255, -1)

op = cv2.bitwise_and(frm, frm, mask=mask) frm[:,
:, 1] = op[:, :, 1] frm[:,
:, 2] = op[:, :, 2]
frm[:max_y, ml:max_x] = cv2.addWeighted(tools, 0.7, frm[:max_y, ml:max_x], 0.3, 0)

cv2.putText(frm, curr_tool, (270+ml,30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255),
2)
cv2.imshow("paint app", frm)


if cv2.waitKey(1) == 27: cv2.destroyAllWindows(
) cap.release() break
```

CHAPTER 7
CONCLUSION AND FUTURE WORK

CONCLUSION

Even senior citizens or people who find it difficult to use keyboards can Effortlessly use the system. Extending the functionality, this system can also be used to control IoT devices shortly. Drawing in the air can also be made possible. This system will be an excellent software for smart wearables using which people could better interact with the digital world . 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.

FUTURE WORK

This system has the potential to challenge traditional writing methods. It eradicates the need to carry a mobile phone in hand notes, providing a simple on-the-go way to do the same. It will also serve a great purpose in helping especially abled people communicate easily, In the future, advancements in Artificial Intelligence will enhance the efficiency of air-writing. This work can be further improved by experimenting with different interpolation methods such as PyGame which includes a line drawing method that could help produce smoother and cleaner lines. In the same vein, a variety of brush shapes and textures can be implemented to make this application more robust.

APPENDIX 1

PUBLICATION DETAILS

We **R.ARUNASALAM, T. BHARATHI, S.BRINDHA, K. SIVARANJANI** Of **K.L.N College of Information Technology** presented the paper **entitled " AIR CANVAS USING HAND GESTURES"** in One Day Virtual National Conference on **"EMERGING TRENDS IN ENGINEERING SCIENCE & TECHNOLOGY "ETEST-23"** organized by **"HINDUSTHAN INSTITUTE OF TECHNOLOGY"** on**20th April2023**

HITECH

HINDUSTHAN INSTITUTE OF TECHNOLOGY

Coimbatore - 641032

(An Autonomous Institution)

Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai, Accredited with 'A' Grade by NAAC & Accredited by NBA)

DEPARTMENT OF MECHANICAL ENGINEERING

&

Indian Society for Technical Education

This is to certify that Mr/Ms/Dr S. BRINDHA of K.I.N. COLLEGE OF INFORMATION TECHNOLOGY. has presented the technical paper entitled AIR CANVAS USING HAND GESTURES. in the **ISTE** sponsored two days National Level Conference on "**Emerging Trends in Engineering Science & Technology "ETEST-23"** during 19th - 20th April 2023

Prof.M.Viswanath
Co-ordinator

Dr.K.M.Arunraja
Organizing Secretary

Hinduslhan Institute of Technology
Department of Mechanical Engineering
Coimbatore - 32

Dr.S.R.RajaBalayanan
Convener & Head

HINDUSTHAN INSTITUTE OF TECHNOLOGY
Coimbatore - 641032
(An Autonomous Institution)
Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai,
Accredited with 'A' Grade by NAAC & Accredited by NBA)

DEPARTMENT OF MECHANICAL ENGINEERING
&
Indian Society for Technical Education

This is to certify that Mr/Ms/Dr K. SIVARANJANI of K.L.N COLLEGE OF INFORMATION TECHNOLOGY. has presented the technical paper entitled AIR CANVAS USING HAND GESTURES in the **ISTE** sponsored two days National Level Conference on "**Emerging Trends in Engineering Science & Technology "ETEST-23"** during 19th- 20th April 2023

Prof.M.Viswanath	**Dr.K.M.Arunraja**	**Dr.S.R.RajaBalayanan**
Co-ordinator	Organizing Secretary	Convener & Head

HINDUSTHAN INSTITUTE OF TECHNOLOGY
Coimbatore - 641032
(An Autonomous Institution)
Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai,
Accredited with 'A' Grade by NAAC & Accredited by NBA)

DEPARTMENT OF MECHANICAL ENGINEERING
&
Indian Society for Technical Education

This is to certify that Mr/Ms/Dr T. BHARATHI of K.L.N COLLEGE OF INFORMATION TECHNOLOGY. has presented the technical paper entitled AIR CANVAS USING HAND GESTURES. in the **ISTE** sponsored two days National Level Conference on "**Emerging Trends in Engineering Science & Technology "ETEST-23"** during 19th- 20th April 2023

Prof.M.Viswanath	**Dr.K.M.Arunraja**	**Dr.S.R.RajaBalayanan**
Co-ordinator	Organizing Secretary	Convener & Head

HINDUSTHAN INSTITUTE OF TECHNOLOGY

Coimbatore - 641032

(An Autonomous Institution)

Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai, Accredited with 'A' Grade by NAAC & Accredited by NBA)

DEPARTMENT OF MECHANICAL ENGINEERING

&

Indian Society for Technical Education

This is to certify that Mr/Ms/Dr R. ARUNASALAM of K.L.N. COLLEGE OF INFORMATION TECHNOLOGY. has presented the technical paper entitled AIR CANVAS USING HAND GESTURES. in the **ISTE** sponsored two days National Level Conference on "**Emerging Trends in Engineering Science & Technology "ETEST-23"** during 19th- 20th April 2023

Prof.M.Viswanath
Co-ordinator

Dr.K.M.Arunraja
Organizing Secretary

Dr.S.R.RajaBalayanan
Convener & Head

REFERENCES

[1] Siam, Sayem & Sakel, Jahidul & Kabir, Md. . Human Computer Interaction Using Marker Based Hand Gesture Recognition., 2016

[2] M. Lee and J. Bae, "Deep Learning Based Real-Time Recognition of Dynamic Finger Gestures Using a Data Glove," in IEEE Access, vol. 8, pp. 219923-219933, 2020, doi: 10.1109/ACCESS.2020.3039401.

[3] S. Khan, M. E. Ali, S. Das and M. M. Rahman, "Real Time Hand Gesture Recognition by Skin Color Detection for American Sign Language," 2019 4th International Conference on Electrical Information and Communication Technology (EICT), 2019, pp. 1-6, doi: 10.1109/EICT48899.2019.9068809.

[4] P. Ramasamy, G. Prabhu and R. Srinivasan, "An economical air writing system converting finger movements to text using web camera," 2016 International Conferenceon Recent Trends in Information Technology (ICRTIT), 2016, pp. 1-6, doi: 10.1109/ICRTIT.2016.7569563.

[5] P. Ramasamy, G. Prabhu and R. Srinivasan, "An economical air writing system converting finger movements to text using web camera," 2016 International Conferenceon Recent Trends in Information Technology (ICRTIT), 2016, pp. 1-6, doi:10.1109/ICRTIT.2016.7569563.

[6] S. Belgamwar and S. Agrawal, "An Arduino Based Gesture Control System for Human- Computer Interface," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, pp. 1-3, doi:10.1109/ICCUBEA.2018.8697673.

[7] Y. Wu and C. -M. Wang, "Applying hand gesture recognition and joint tracking to a TV controller using CNN and Convolutional Pose Machine," 2018 24th International Conference on Pattern Recognition (ICPR), 2018, pp. 3086-3091, doi: 10.1109/ICPR.2018.8546209.

[8] Q. De Smedt, H. Wannous and J. Vandeborre, "Skeleton-Based Dynamic Hand GestureRecognition," 2016 IEEE Conference on Computer Vision and Pattern Recognitio Workshops (CVPRW),20161206-1214,doi: 10.1109/CVPRW.2016.153.

[9] Prajakta Vidhate, Revati Khadse, Saina Rasal, “Virtual paint application by hand gesture recognition system”, 2019 International Journal of Technical Research and Applications, 2019, pp. 36-39.

[10] R. Lyu et al., "A flexible finger-mounted airbrush model for immersive freehand painting," 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), 2017, pp. 395-400, doi:10.1109/ICIS.2017.7960025.

[11] Y. Huang, X. Liu, X. Zhang, and L. Jin, "A Pointing Gesture Based Egocentric Interaction System: Dataset, Approach, and Application," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, pp. 370-377, 2016.

[12] P. Ramasamy, G. Prabhu, and R. Srinivasan, "An economical air writing system is converting finger movements to text using a web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, pp. 1- 6,2016.

[13] Saira Beg, M. Fahad Khan and Faisal Baig, "Text Writing in Air," Journal of Information Display Volume 14, Issue 4, 2013

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

[15] Yuan-Hsiang Chang, Chen-Ming Chang, "Automatic Hand-Pose Trajectory TrackingSystem Using Video Sequences", INTECH, pp. 132- 152,

Croatia, 2010 Erik B. Sudderth, Michael I. Mandel, William T. Freeman, Alan S. Willsky

[16] "VisualHand Tracking Using Nonparametric Belief Propagation", MIT Laboratory For Information & Decision Systems Technical Report P-2603, Presented atIEEE CVPR Workshop On Generative Model-Based Vision, Pp. 1-9, 2004

[17] T. Grossman, R. Balakrishnan, G. Kurtenbach,

G. Fitzmaurice, A. Khan, and B. Buxton, "Creating Principal 3D Curves with Digital TapeDrawing," Proc. Conf. Human Factors Computing Systems (CHI' 02), pp. 121-128, 2002.

[18] T. A. C. Bragatto, G. I. S. Ruas, M. V. Lamar, "Real-time Video-Based Finger Spelling Recognition System Using Low Computational Complexity Artificial NeuralNetworks",

IEEE ITS, pp. 393-397, 2006

[19] 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 Computational Intelligence, Brisbane, Australia, 2012 [20