INTRODUCTION

1.0 INTRODUCTION

Sketching On Air is possible through our trending technology namely open cv, python. Open cv is mainly known as an open-source computer vision and machine learning software. The library has more than 2400 best algorithms, which includes comprehensive set of classic and state-of-the-art computer vision and machine learning algorithms. Most of these algorithms are used to detect and recognize faces, identify objects, classify human activities in videos track camera movements, track moving objects, extract 3D one's.

Python is one of the high-level-general-purpose programming language. Object-oriented approach mainly to help programmers to write clear, logical code for small as well as large – scale projects. 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.

Virtual AI painter using OpenCV and Mediapipe is an application that tracks the movement of an object. Using this tracking feature, the user can draw on the screen by moving the object (which in our project is the human hand) in the air, in front of the webcam. This real time webcam data generated by tracking the movement of the object helps the user to draw simple things which are both interesting and challenging. OpenCV (Open-Source Computer Vision) - is a programming language library consisting of different types of functions mainly for computer vision. To explain in a simple language or in general way it is a library used for image processing. It is used mainly to do all the operations which are related to images.

What it can do:

1. Read and write images.
2. Detection of faces and its features.
3. Detection of different shapes such as circle, rectangle etc. in an image. E.g. Detection of coins in images.
4. Text recognition in images. e.g., Reading Number Plates.
5. Can modify the quality of an image or its color.
6. Developing Augmented reality apps.

1.1 OPENCV

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

It's the basic introduction to OpenCV we can continue the Applications and all the things in our upcoming articles.

1.1.1 Applications of OpenCV:

There are lots of applications which are solved using OpenCV, some of them are listed below

- face recognition
- Automated inspection and surveillance
- Vehicle counting on highways along with their speeds
- Interactive art installations
- Street view image stitching
- Video/image search and retrieval
- object recognition
- Medical image analysis
- Movies – 3D structure from motion
- TV Channels advertisement recognition

1.1.2 OpenCV Functionality:

- Image/video I/O, processing, display
- Object/feature detection
- Geometry-based monocular or stereo computer vision
- Computational photography
- Machine learning & clustering

1.2 MEDIAPIPE

MediaPipe is Google's open-source framework (graph based) used for media processing. Mainly aims at making media processing easier for us by providing machine learning features and some integrated computer vision.

Some of its notable applications are as follows:

1. Face detection
2. Multi hand tracking

2.SYSTEM ANALYSIS

2.0 SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analysts firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

The existing system works with your fingers only, and there are no highlighters, paints, or relatives. Identifying and characterizing an object such as a finger from an RGB image without a depth sensor is a great challenge. Another problem is the lack of up and down movement of the pen. The system uses a single RGB camera to write from above. Since depth detection is not possible, the up and down activities of the pen cannot be tracked. Therefore, the entire path of the fingertip is drawn, and the resulting image would be absurd and not recognized by the model. Using real-time hand gestures to change the system from one state to another requires much code care. In addition, the user must know many movements to control his plan adequately. The project focuses on solving some critical social problems.

2.2 EXISTING SYSTEM

Your screen is a device that does exactly what you are asking - it displays data. There are various ways to achieve this, one being the keyboard which is a traditional and widely used method to display data on the screen, it has a keyboard composed of buttons used to create letters, numbers, and symbols, and perform additional functions but it has a few disadvantages such as It is a slow method when you need to write a long piece of writing when there are faster ways such as scanning and dictation.

The second method is the speech to text, this is a software that works by listening to audio and delivering an editable, verbatim transcript on a given device. The software does this through voice recognition. The drawbacks of this method is that the voice recognition software won't always put your words on the screen completely accurately. Programs cannot understand the context of language the way that humans can, leading to errors that are often due to misinterpretation and it cannot always differentiate between homonyms. Another method is a touchscreen. It is a computer screen that can be used by touching it with a finger or a stylus pen, instead of using a mouse and keyboard. It can be described as a touchpad with a screen built-in to it. A few disadvantages are that they are not suitable for inputting large amounts of data, not very accurate - selecting detailed objects can be difficult with fingers it is more expensive than alternatives such as a mouse and can soon become faulty if misused.

2.2.1 DISADVANTAGES OF EXISTING SYSTEM

- Difficulty with software Development - slow and Expansive.
- It is a slow method when you need to write a long piece of writing when there are faster ways such as scanning and dictation.

2.3 PROPOSED SYSTEM

Drawing or Sketching using hand is everyone's wish. Some or the other time we imagine writing in air using our hand. So, here came the project from this concept where we create a canvas and pick the colors required using our hand and draw the required design or write anything you wish. Gestures are non-verbal information used to improve computer language understanding. Human gestures are perceived via sight, and computer vision is used to research different gestures. The assignment takes gain of this shortcoming and makes a specialty of developing a motion to text converter able to serving as software for clever wearables to document from the air.

The device will use computer vision to music the path of the finger, and on this manner, you will write from the top down. The generated text also can be used for extraordinary purposes, which includes sending messages, emails, etc. it is going to be a powerful approach of communique for the listening to impaired. it's far a powerful approach of communique that gets rid of the need to put in writing and reduces using cell telephones and laptops. The library has more than 2400 best algorithms, which includes comprehensive set of classic and state-of-the-art computer vision and machine learning algorithms.

Most of these algorithms are used to detect and recognize faces, identify objects, classify human activities in videos track camera movements, track moving objects, extract 3D one's. Python is one of the high-level-general- purpose programming language. Object-oriented approach mainly to help programmers to write effectively as well as large scale.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- It is an alternative for advanced teaching software, it could be used by anyone having no prior knowledge about computer, being cost effective it could be used by teachers in remote area as well for making their videos more explanatory.
- Easier tracing.
- Don't need a specific software.

2.4 HARDWARE & SOFTWARERE QUIREMENTS

2.4.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system.

The following are some hardware requirements.

- Processor: IntelDualCoreI3and above
- Hard disk: 8GBand above
- RAM: 8GBand above
- Input devices: Keyboard, mouse.

2.4.2 SOFTWARERE QUIREMENTS:

Software requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

- Operating system: Windows8 and above
- Languages: Python
- Tools: PythonIDEL3.7version, Visual Studio, google colab

2.5 APPROACH

2.5.1 Writing Hand Pose Detection

Recognizing the position of the composing hand and recognizing it through other signals is a fundamental step in initializing airborne composing. Not at all like conventional writing, when the write moves down, and the write moves up, composing within the discuss isn't laid out as a writing arrangement. Events. The framework recognizes the position of a piece hand and recognizes it from a non-writing hand by tallying the number of raised fingers.

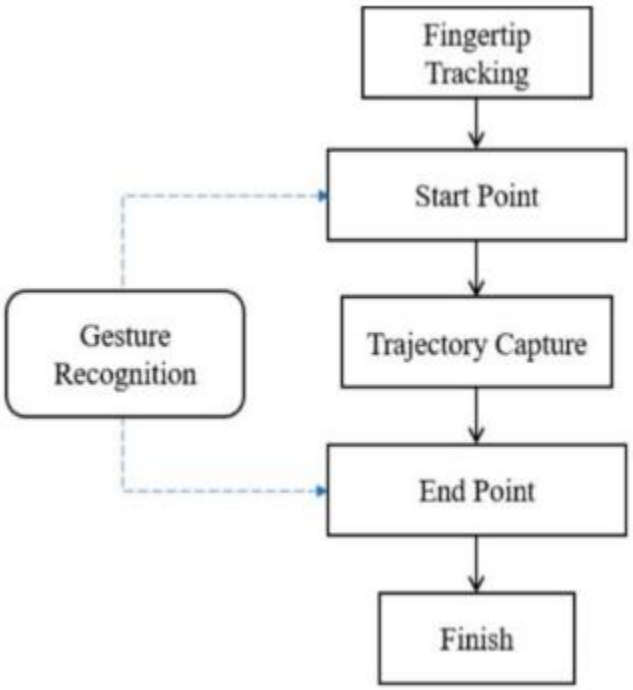


Fig: Flowchart of Proposed System

2.5.2 Hand Region Segmentation

Once we have precisely captured the hand utilizing the over procedure, the division of the hand zone is done employing a two-step approach, viz. The skin division and the subtraction of the foundation and the ultimate parallel picture of the hand are obtained as a conglomeration of the two. The proposed algorithm works well in real-time and gives moderately precise division. In spite of the fact that skin colors shift incredibly from breed to breed, it has been watched that skin color features a little region between distinctive skin sorts, whereas skin luminosity differs significantly.

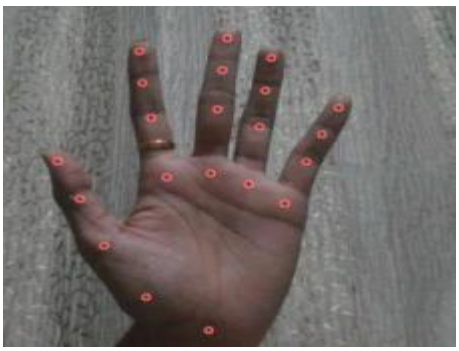


Fig: Hand Region Segmentation

Background subtraction: Since precise hand location with the Speedier R-CNN handheld locator taken after by sifting of skin color at the boundary of the candidate's hand gives a sensibly great division result, the subtraction step of background is as it were utilized to evacuate skin-colored objects (not portion of the hand) that are within the bounding box of the recognized hand may be present.

2.5.3 Hand Centroid Localization

Since deciding the precise center of gravity of the hand is basic within the taking after steps, the framework employments two calculations to decide the initial estimates of the center of gravity, and the ultimate center of gravity is calculated as the normal of the two. The remove change strategy is utilized to obtain the first assess of the center of gravity (xc1, yc1). Within the separate change picture, each pixel is spoken to by its separate from the following edge pixel. Euclidean separate was utilized to degree the distance between a pixel and its closest edge pixel. Hence, the pixel with the most noteworthy escalated within the separate change picture is taken as the center of gravity.

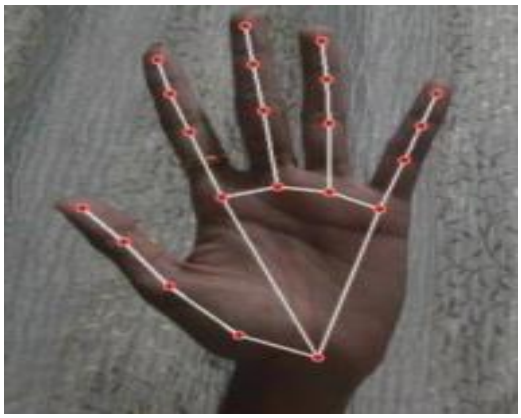


Fig: Hand Centroid Localization

2.5.4 Drawing Line using position of the contour

The real rationale behind this computer vision extend is to form a Python deque (an information structure). The deque will memorize the position of the diagram in each subsequent outline, and we are going utilize these collected focuses to form a line utilizing OpenCV's drawing capabilities. Presently utilize the layout position to choose whether to press a button or draw on the given sheet. A few of the buttons are at the best of the canvas. When the pointer enters this range, it is enacted agreeing to the strategy show in this area.

Fig: Flowchart of Proposed System

Object position: Picture extricated from the video arrangement. Extricate the color picture from the reference: This proposed strategy tracks the file finger development, colored blue. We do not have a reference picture, so any past picture may be a reference to the another. Presently take the contrast within the pictures and extricate the color and development of the object. Edge Upgrade (EE) - Edge upgrade procedure makes the question area calculation strong against clamor, diverse lighting conditions, obscuring, and blurring of objects, indeed in moo differentiate images.

2.6 MODULES

Below mentioned modules implemented in our project:

1) Hand Land Mark Detection:

It is a process of finding joints on the finger as well as the finger tips on a image. Basically, this module produce 22 key points of which 21 key points are on the hand/palm, and the 22nd point is considered as the background.

2) Drawing Tools Module:

In this we created function for selecting tools such as curve, straight line, circle, rectangle and eraser. In addition to above module, we have also added audio module for which we can get audio of our selected tool but it has adverse effect on system with lower configuration but we very much sure that it would fine with the higher configuration system.

3.ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure for how the procedure done during the project execution.

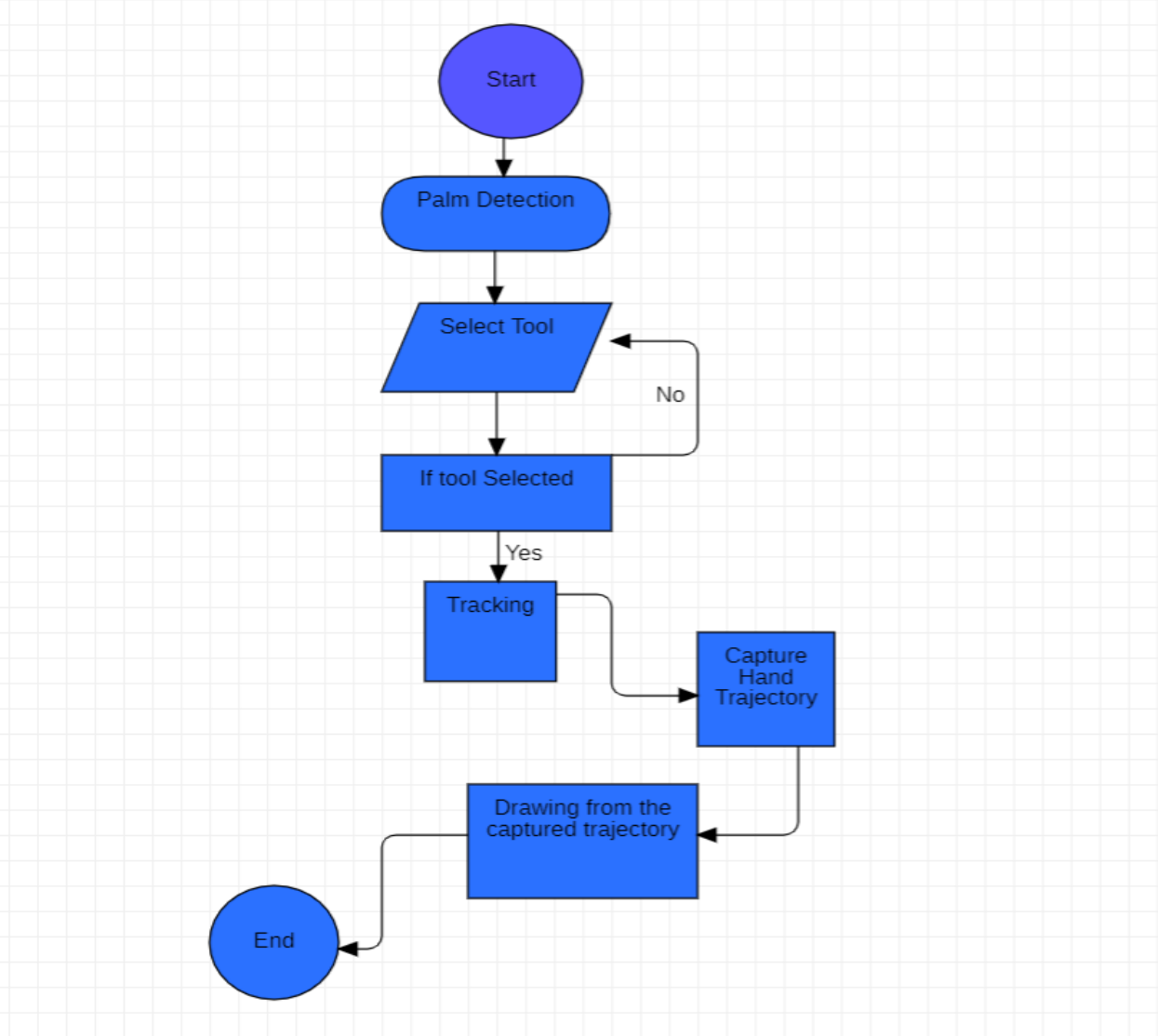


Figure3.1: Project Architecture of Computer Vision Based Virtual Sketch Using OpenCV

3.2 ALGORITHM DIAGRAM

Algorithm for Program Execution procedure

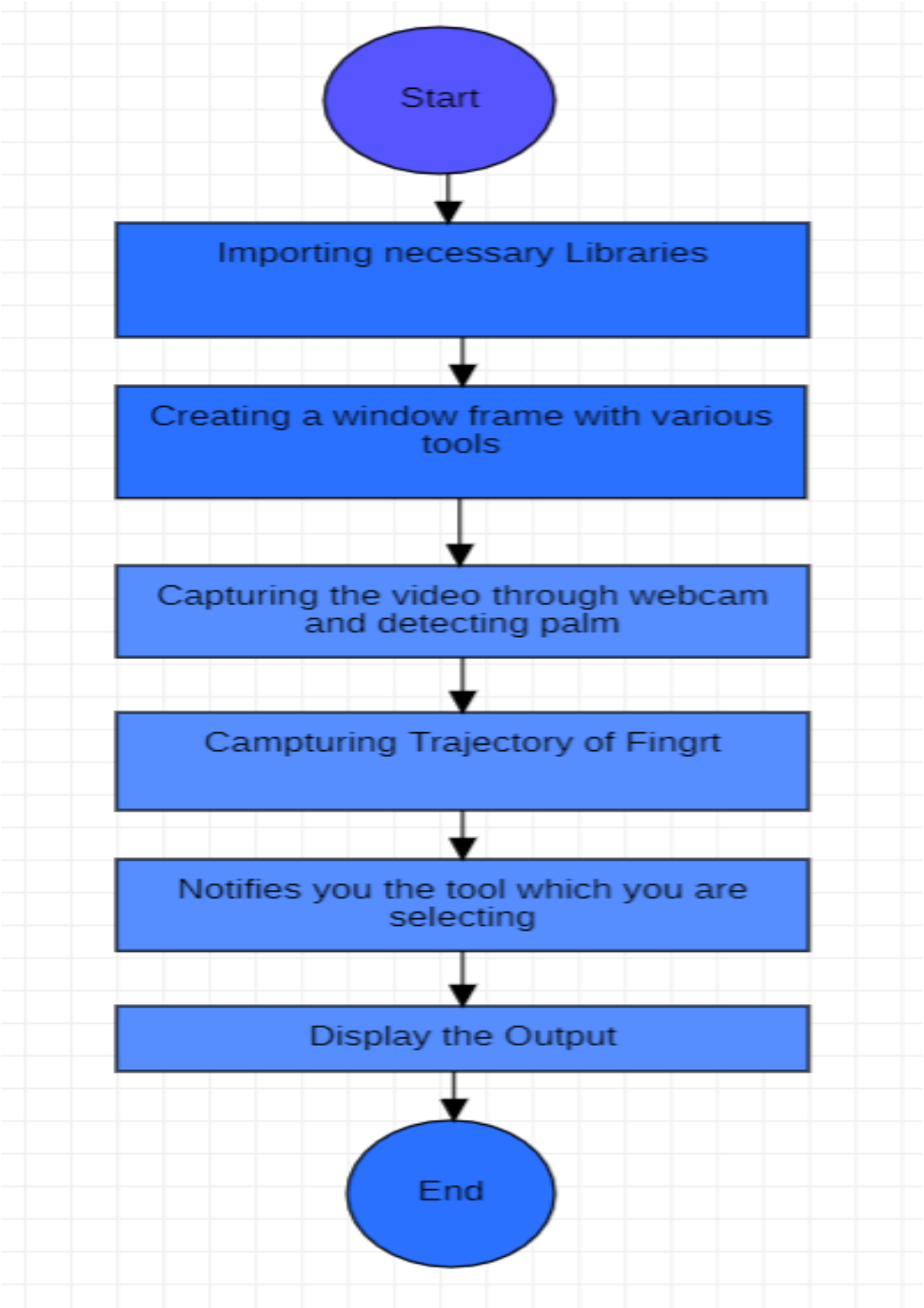


Figure3.2: Project Architecture of Computer Vision Based Virtual Sketch Using OpenCV

3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model. A use case diagram isa graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

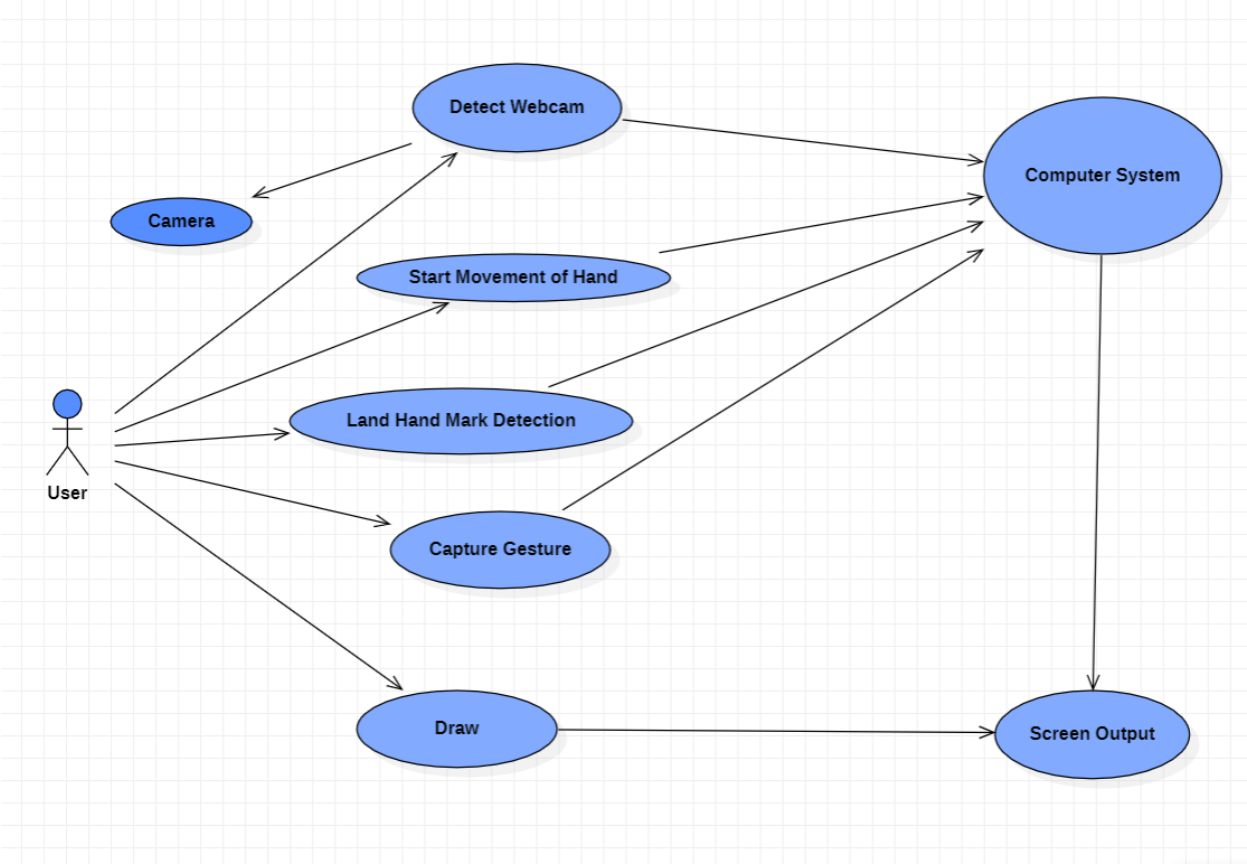


Figure3.3: Use Case Diagram for Computer Vision Based Virtual Sketch Using OpenCV

3.4 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

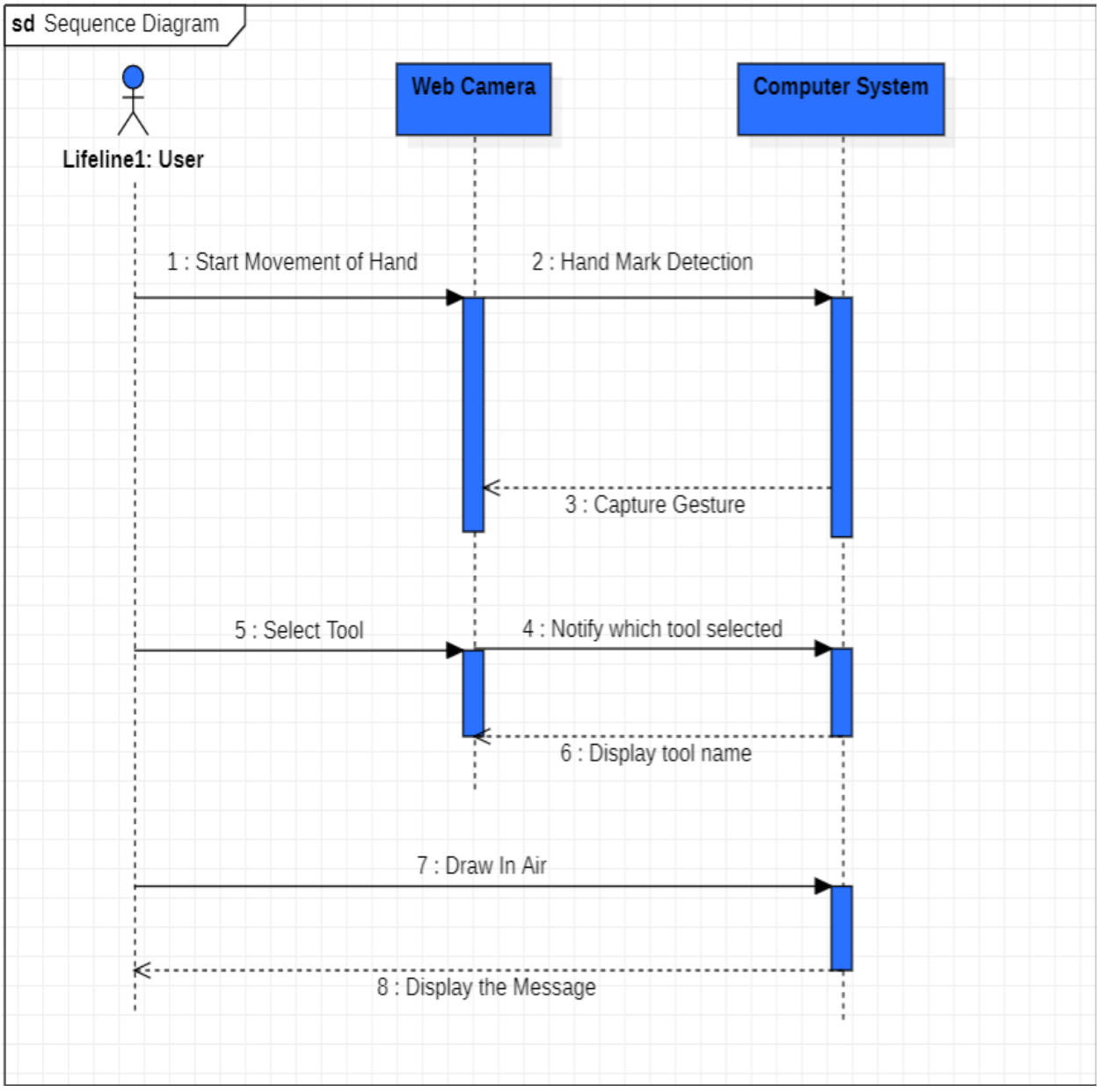


Figure3.4: Sequence Diagram for Computer Vision Based Virtual Sketch Using OpenCV

3.5 Class Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

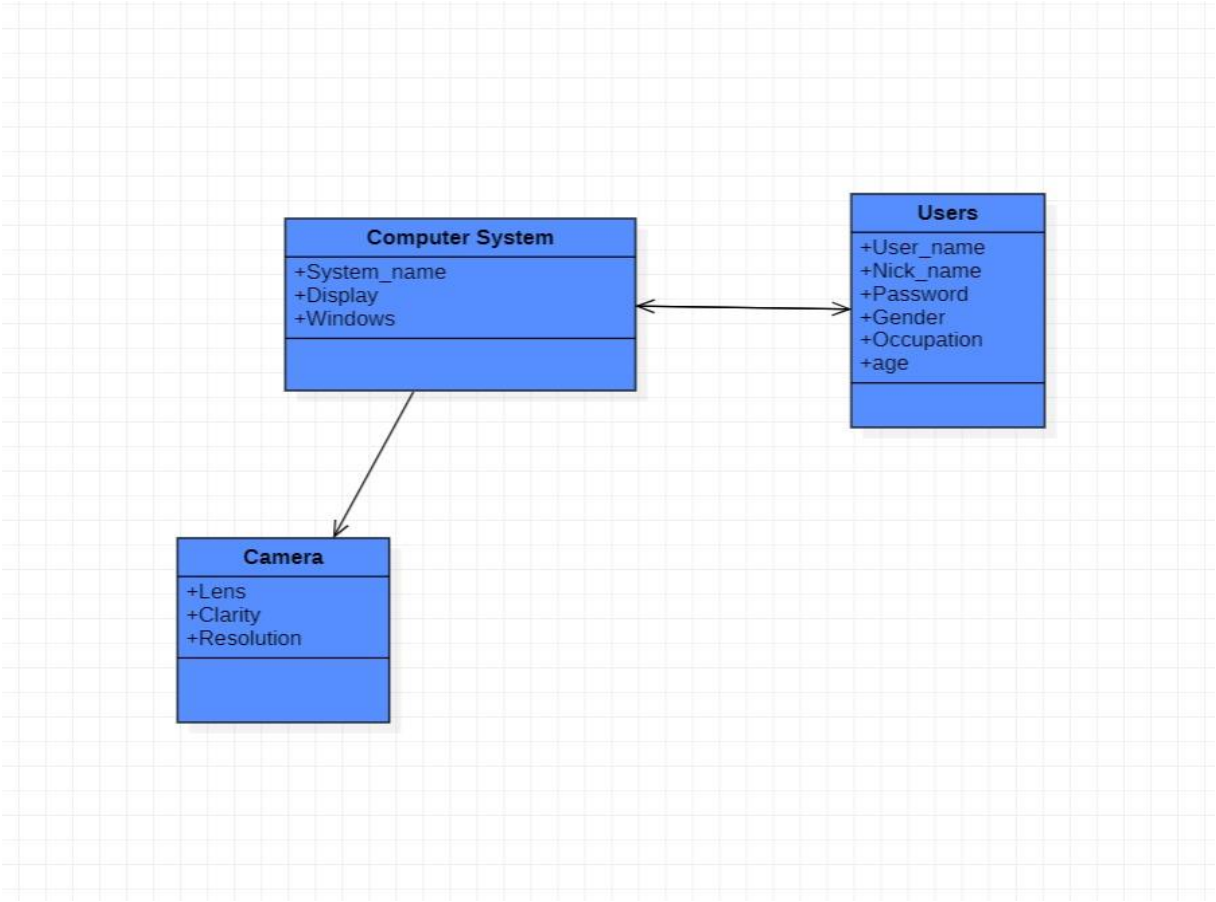


Figure3.4: Class Diagram for Computer Vision Based Virtual Sketch Using OpenCV

4. IMPLEMENTATION

4.1 SAMPLE CODE

```
# pip install mediapipe

# pip install python-math

# pip install numpy

# pip install opencv-python

# pip install python-time

# pip install os-sys

# pip install Hand-Tracking-Module

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('C:/Users/HP/Desktop/collage project/major project/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
elifcurr_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
elifcurr_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

elifcurr_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
```

5. SCREENSHOT

Screenshot 5.1: Showing the tools in computer system

Screenshot 5.2: Showing hand gestures

Screenshot 5.3: Showing single figure to select

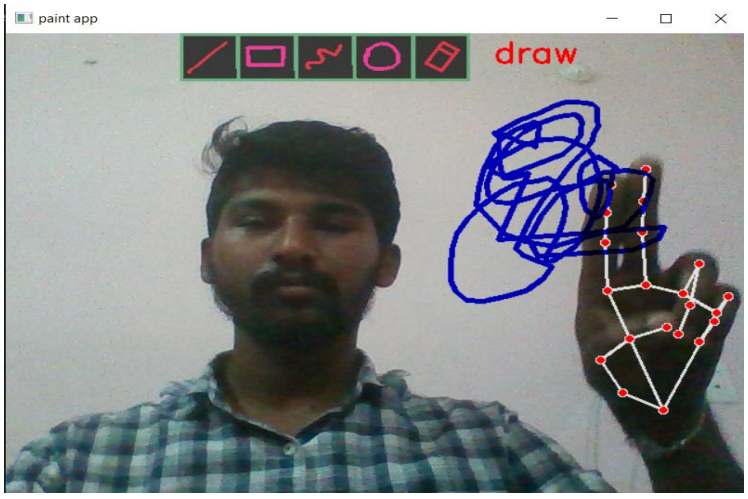
Screenshot 5.4: Showing two figure to write

Screenshot 5.5: Selecting erase from computer system

Screenshot 5.6: Selecting Line from computer system

Screenshot 5.7: Selecting rectangle from computer system

Screenshot5.8: Selecting circle from computer system



Screenshot 5.9: Selecting draw from computer system

6.TESTING

6.0 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.1 TYPES OF TESTING

6.1.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.1.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.1.3 TOP-DOWN INTEGRATION

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner. In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

6.1.4 BOTTOM-UP INTEGRATION

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom-up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program Structure The bottom-up approaches test each module individually and then each module is module is integrated with a main module and tested for functionality.

6.1.5 USER ACCEPTANCE TESTING

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

6.1.6 OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

This system has the potential to challenge traditional writing methods. It eradicates the need to carry a mobile phone in hand to jot down 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. 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. Augmented reality can make text come alive. 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 OpenCV 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.

7.2 FUTURE SCOPE

This system could be used as an alternative for teaching software used by teachers. If further interpreted various virtual based physical games could be made. Controlling the robot using gestures considered as one of the interesting applications in this field proposed a system controlling a robot using hand pose signs. The orders could be given to robot to execute some tasks, where each sign has a specific meaning and represents different function.

8. BIBLIOGRAPHY

8.1 REFERENCES:

- [1] S Guennouni, A Ahaitouf and A Mansouriss , “Multiple object detection using Open CV on an embedded platform”, 2014 Third IEEE International Colloquium in Information Science and Technology (CIST), 2014, pp. 374-377.
- [2] G. Chandan, Mohana A.H Jain “The Real Time Object Detection and Tracking Using Deep Learning and OpenCV”, 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 2018, pp. 1305-1308.
- [3] Prof. S.U. Sahoj, Nishtha Dua, AkashKumar Choudhary, Bharat Phogat air canvas application using OpenCV and NumPy in python.
- [4] Siddharth Mandgi1, Shubham Ghatge1, Mangesh Khairnar1, Kunal Gurnani1, Prof. Amit Hatekar2 Object Detection And Tracking Using Image Processing Vol. 8, Issue 2, (Part -1) February 2018, pp.39-41
- [5] Hemalatha Vadlamudi, Evaluation of Object Tracking System using Open-CV In Python International Journal of Engineering Research & Technology (IJERT)
- [6] Saira Beg, M. Fahad Khan and Faisal Baig, "Text Writing in Air," Journal of Information Display Volume 14.
- [7] 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, 2016, pp. 1- 6.
- [8] 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.
- [8] https://www.geeksforgeeks.org/live-webcam-drawing-using-opencv/

8.2 GITHUBLINK

https://github.com/197R1A0520/Computer-Vision-Based-Virtual-Sketch-Using-OpenCV

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

COMPUTER VISION BASED VIRTUAL SKETCH USING OPENCV

[1] Mr. Punyaban Patel , [2] Keerthi Ruchitha,[3] Kurumula Neelima,

[4] Santhosh Kumar Karamthot

1. *Professor Department of Computer Science and Engineering, CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.*

Email-: punyaban@gmail.com ,Mob:- (+91)7008359109

2. *[2,3,4] B. Tech Students, Department of Computer Science and Engineering, CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.*

Email-: [2] keerthiruchitha@gmail.com, [3] kurumulaneelima8623@gmail.com

[4] karamthotsanthosh123@gmail.com

ABSTRACT- Virtual Sketch is in where we can draw by just capturing the motion of a colored marker with a camera. One colored object at the tip of the finger is mainly used as the marker. We are here now, using the techniques of computer vision in opencv to build this project. The required language for this project is python due to its more exhaustive libraries and easy to make use of the syntax and but understanding the basics as well as it can be implemented in any opencv supported languages The color tracking and detection processes are used to achieve the goal of this project.

Virtual Painting is a canvas-based platform on which we can draw by just motion of the hand Basically it just track the hand & capture the motion of fingers in this process the tip of the fingers are mainly used as the marker. It mainly uses is the opencv technology, which is backbone of Augmented Reality. Virtual Painting is fully developed in Python, it implements the basic and advance levels of python. The color tracking and detection process is used to achieve the output. Here the color marker is used to produce a mask on the original color canvas.

KEYWORDS: Opencv, Camera, Virtual Painting.

ISSN: 2057-5688

1. INTRODUCTION

Sketching On Air is possible through our trending technology namely open cv, python. Open cv is mainly known as an open-source computer vision and machine learning software. The library has more than 2400 best algorithms, which includes comprehensive set of classic and state-of-the-art computer vision and machine learning algorithms. Most of these algorithms are used to detect and recognize faces, identify objects, classify human activities in videos track camera movements, track moving objects, extract 3D one's. Python is one of the high-level-general-purpose programming language. Objectoriented approach mainly to help programmers to write clear, logical code for small as well as large – scale projects. 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. Virtual AI painter using OpenCV and Mediapipe is an application that tracks the movement of an object. Using this tracking feature, the user can draw on the screen by moving the object (which in our project is the human hand) in the air, in front of the webcam. This real time webcam data generated by tracking the movement of the object helps the user to draw simple things which are both interesting and challenging. OpenCV (Open-Source Computer Vision) - is a programming language library consisting of different types of functions mainly for computer vision. To explain in a simple language or in general way it is a library used for Image Processing. It is used mainly to do all the operations which are related to Images.

Applications of OpenCV: There are lots of applications which are solved using OpenCV, some of them are listed below

face recognition

Automated inspection and surveillance

Vehicle counting on highways along with

ISSN: 2057-5688

their speeds

Interactive art installations

Street view image stitching

Video/image search and retrieval

Object recognition

Medical image analysis

Movies – 3D structure from motion

TV Channels advertisement recognition

2.SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, "what must be done to solve the problem?" The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analysts firm understanding of what is to be done.

PROBLEM DEFINITION:

The existing system works with your fingers only, and there are no highlighters, paints, or relatives. Identifying and characterizing an object such as a finger from an RGB image without a depth sensor is a great challenge. Another problem is the lack of up and down movement of the pen. The system uses a single RGB camera to write from above. Since depth detection is not possible, the up and down activities of the pen cannot be tracked. Therefore, the entire path of the fingertip is drawn, and the resulting image would be absurd and not recognized by the model. Using real-time hand gestures to change the system from one state to another requires much code care. In addition, the user must know many movements to control his plan adequately. The project focuses on solving some critical social problems.

3.EXISTING SYSTEM:

Your screen is a device that does exactly what you are asking - it displays data. There are various ways to achieve this, one being

ISSN: 2057-5688

the keyboard which is a traditional and widely used method to display data on the screen, it has a keyboard composed of buttons used to create letters, numbers, and symbols, and perform additional functions but it has a few disadvantages such as It is a slow method when you need to write a long piece of writing when there are faster ways such as scanning and dictation.

DISADVANTAGES OF EXISTING SYSTEM:

Difficulty with software Development - slow and Expansive.

• It is a slow method when you need to write a long piece of writing when there are faster ways such as scanning and dictation.

4. PROPOSED SYSTEM:

Drawing or Sketching using hand is everyone's wish. Some or the other time we imagine writing in air using our hand. So, here came the project from this concept where we create a canvas and pick the colors required using our hand and draw the required design or write anything you wish. Gestures are non-verbal information used to improve computer language understanding. Human gestures are perceived via sight, and computer vision is used to research different gestures. The assignment takes gain of this shortcoming and makes a specialty of developing a motion to text converter able to serving as software for clever wearables to document from the air. The device will use computer vision to music the path of the finger, and on this manner, you will write from the top down. The generated text also can be used for extraordinary purposes, which includes sending messages, emails, etc. it is going to be a powerful approach of communique for the listening to impaired. it's far a powerful approach of communique that gets rid of the need to put in writing and reduces using cell telephones and laptops. The library has more than 2400 best algorithms, which includes comprehensive set of classic and state-of-the- art computer vision and machine learning algorithms. Most of these algorithms are used to detect and recognize faces, identify objects, classify human activities in videos track camera movements, track moving objects, extract 3D one's. Python is one of the high-level-general- purpose programming language. Object-oriented approach mainly

ISSN: 2057-5688

to help programmers to write effectively as well as large scale.

ADVANTAGES OF PROPOSED SYSTEM:

It is an alternative for advanced teaching software, it could be used by anyone having no prior knowledge about computer, being cost effective it could be used by teachers in remote area as well for making their videos more explanatory. • Easier tracing. • Don't need a specific software.

5. HARDWARE & SOFTWARE REQUIREMENTS

HARDWAREREQUIREMENTS:

Processor: IntelDualCoreI3and above

- Hard disk: 8GBand above
- RAM: 8GBand above
- Input devices: Keyboard, mouse.

SOFTWAREREQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

- Operating system: Windows8 and above

Languages: Python

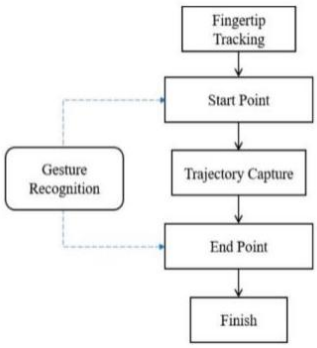
- Tools: Python IDEL3.7version,Visual Studio ,googlecolab .

6. APPROACH

Writing Hand Pose Detection Recognizing the position of the composing hand and recognizing it through other signals is an fundamental step in initializing airborne composing. Not at all like conventional writing, when the write moves down, and the write moves up, composing within the discuss isn't laid out as a writing arrangement. Events. The framework recognizes the position of a piece hand and recognizes it from a non-writing hand by tallying the number of raised fingers. Hand Region Segmentation Once we have precisely captured the hand utilizing the over procedure, the division of the hand zone is done employing a two-step approach, viz. The skin division and the subtraction of the foundation and the ultimate parallel picture of the hand are obtained as a

ISSN: 2057-5688

conglomeration of the two.



The proposed algorithm works well in real-time and gives moderately precise division. In spite of the fact that skin colors shift incredibly from breed to breed, it has been watched that skin color features a little region between distinctive skin sorts, whereas skin luminosity differs significantly.

Fig: Hand Centroid Localization

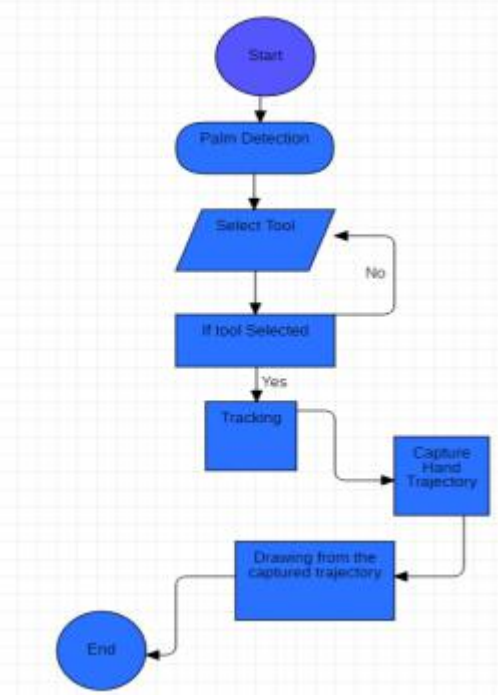
Fig: Flowchart of Proposed System

7. ARCHITECTURE

PROJECT ARCHITECTURE: This project architecture shows the procedure for how the procedure done during the project

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688

execution



Screenshot5.6: Selecting Line from computer system

8.RESULT

Screenshot5.1: Showing the tools in computer system

Screenshot5.2: Showing hand gestures

9. CONCLUSION

This system has the potential to challenge traditional writing methods. It eradicates the need to carry a mobile phone in hand to jot down 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. 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

ISSN: 2057-5688

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. Augmented reality can make text come alive. 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 OpenCV 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.

10. FUTURESCOPE

This system could be used as an alternative for teaching software used by teachers. If further interpreted various virtual based physical games could be made. Controlling the robot using gestures considered as one of the interesting applications in this field proposed a system controlling a robot using hand pose signs. The orders could be given to robot to execute some tasks, where each sign has a specific meaning and represents different function.

8. REFERENCES

• [1] S Guennouni, A Ahaitouf and A Mansouriss , “Multiple object detection using Open CV on an embedded platform”, 2014 Third IEEE International Colloquium in Information Science and Technology (CIST), 2014, pp. 374-377.

• [2] G. Chandan, Mohana A.H Jain “The Real Time Object Detection and Tracking Using Deep Learning and OpenCV”, 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 2018, pp. 1305-1308.

•[3]https://ieeexplore.ieee.org/abstract/document/9742419

•[4]https://ieeexplore.ieee.org/abstract/document/9382703

•[5]https://ieeexplore.ieee.org/abstract/document/9340297

ISSN: 2057-5688

•[6]https://ieeexplore.ieee.org/abstract/document/9641587

•[7]https://ieeexplore.ieee.org/abstract/document/9753939

•[8]https://ieeexplore.ieee.org/abstract/document/9775918

GITHUBLINK

https://github.com/197R1A0520/Computer-Vision-Based-Virtual-Sketch-Using-OpenCV

Keerthi Ruchitha is currently pursuing B. Tech final year in the stream of Computer Science and Engineering in CMR Technical Campus, Medchal, Hyderabad, Telangana, India.

Kurumula Neelima is currently pursuing B. Tech final year in the stream of Computer Science and Engineering in CMR Technical Campus, Medchal, Hyderabad, Telangana, India.

Santhosh Kumar Karamthot is currently pursuing B. Tech final year in the stream of Computer Science and Engineering in CMR Technical Campus, Medchal, Hyderabad, Telangana, India.

Mr.Punyaban Patel is currently working as an Professor in the

ISSN: 2057-5688

Department of Computer Science and Engineering, CMR Technical Campus, Kandlakoya, Medchal, Hyderabad. His areas of Specialization include Machine Learning, Data Processing, Image Processing. He has more than 22 years of Teaching Experience.

INTERNATIONAL JOURNAL OF TECHO-ENGINEERING

International Journal of Techno-Engineering (IJTE)

UGC CARE APPROVED - ESTD Year: 2009

ISSN NO : 0975-4520 | web : http://ijte.uk | e-mail : ijte@gmail.com

CERTIFICATION OF PUBLICATION

This is to certify that the paper entitled

"COMPUTER VISION BASED VIRTUAL SKETCH USING OPENCV"

Authored by:

Keerthi Ruchitha

From

CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.

Has been published in

IJTE JOURNAL, VOLUME XV, ISSUE I, January - March - 2023

UGC APPROVED JOURNAL

Sravan

D Sravan Kumar Editor-In-Cheif **IJET**

5.3

IMPACT FACTOR

INTERNATIONAL JOURNAL OF TECHO-ENGINEERING

International Journal of Techno-Engineering (IJTE)

UGC CARE APPROVED - ESTD Year: 2009

ISSN NO : 0975-4520 | web : http://ijte.uk | e-mail : ijte@gmail.com

CERTIFICATION OF PUBLICATION

This is to certify that the paper entitled

"COMPUTER VISION BASED VIRTUAL SKETCH USING OPENCV"

Authored by:

Kurumula Neelima

From

CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.

Has been published in

IJTE JOURNAL, VOLUME XV, ISSUE I, January - March - 2023

UGC APPROVED JOURNAL

D Sravan Kumar Editor-In-Cheif **IJET**

5.3

IMPACT FACTOR

INTERNATIONAL JOURNAL OF TECHO-ENGINEERING

International Journal of Techno-Engineering (IJTE)

UGC CARE APPROVED - ESTD Year: 2009

ISSN NO : 0975-4520 | web : http://ijte.uk | e-mail : ijte@gmail.com

CERTIFICATION OF PUBLICATION

This is to certify that the paper entitled

"COMPUTER VISION BASED VIRTUAL SKETCH USING OPENCV"

Authored by:

Santhosh Kumar Karamthot

From

CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.

Has been published in

IJTE JOURNAL, VOLUME XV, ISSUE I, January - March - 2023

UGC APPROVED JOURNAL

D Sravan Kumar Editor-In-Cheif **IJET**

5.3

IMPACT FACTOR

INTERNATIONAL JOURNAL OF TECHO-ENGINEERING

International Journal of Techno-Engineering (IJTE)

UGC CARE APPROVED - ESTD Year: 2009

ISSN NO : 0975-4520 | web : http://ijte.uk | e-mail : ijte@gmail.com

CERTIFICATION OF PUBLICATION

This is to certify that the paper entitled

"COMPUTER VISION BASED VIRTUAL SKETCH USING OPENCV"

Authored by:

Mr. Punyaban Patel, Professor

From

CMR TECHNICAL CAMPUS Kandlakoya(V) , Medchal(M), Hyderabad (Pin:501401), Telangana, India.

Has been published in

IJTE JOURNAL, VOLUME XV, ISSUE I, January - March - 2023

UGC APPROVED JOURNAL

D Sravan Kumar Editor-In-Cheif **IJET**

5.3

IMPACT FACTOR