# Mini Project Title

Submitted in partial fulfillment of the requirements of the degree

**BACHELOR OF ENGINEERING** IN **COMPUTER ENGINEERING**

By

**Nishita Matlani 1902096**

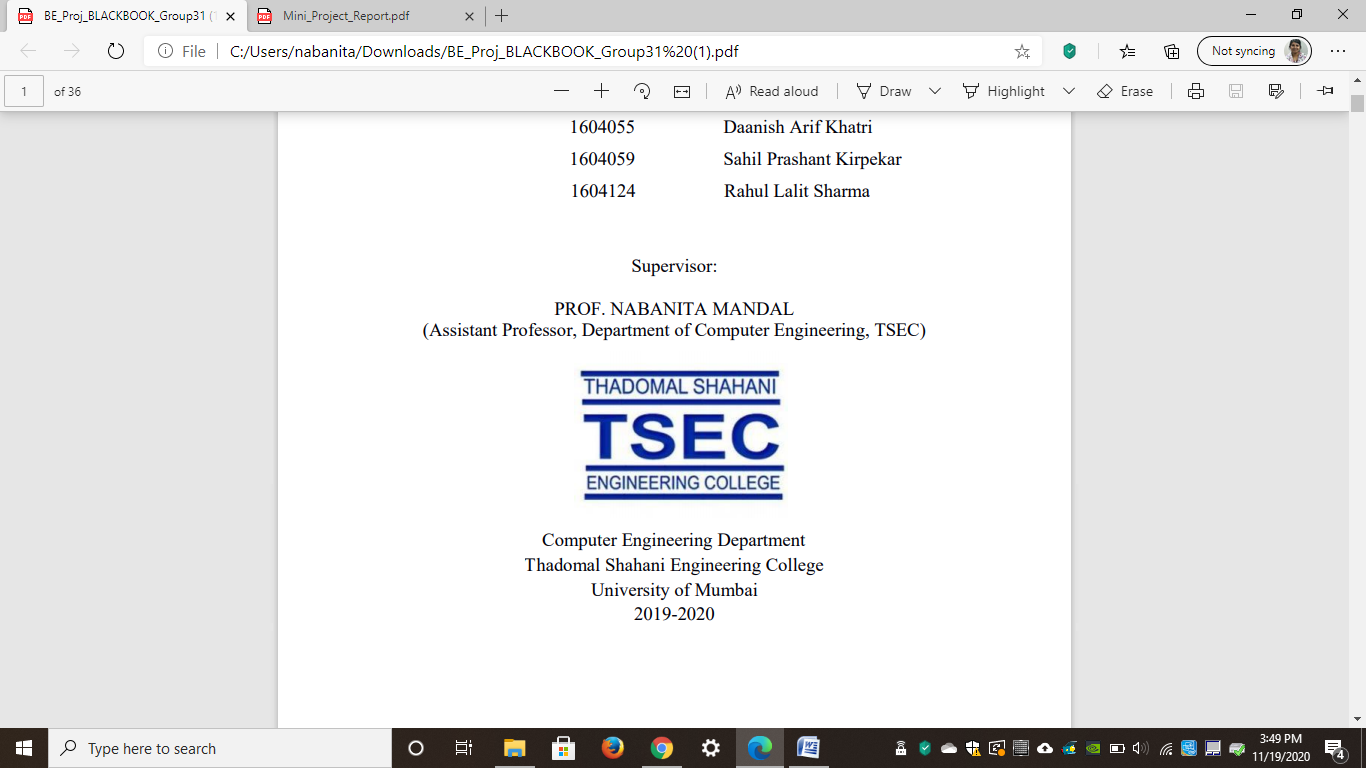
**Isha Mehta 1902099**

**Shanvi Mehta 1902100**

**Harsh Mehta 1902098**

**Dr. Jayant Gadge**

Supervisor Name



**Computer Engineering Department**

**Thadomal Shahani Engineering College**

**Bandra(w), Mumbai - 400 050**

**University of Mumbai**

**(AY 2020-21)**

# CERTIFICATE

This is to certify that the Mini Project entitled **“ Finger Tarp ”** is a bonafide work of **Nishita Matlani (1902096), Isha Mehta (1902099), Shanvi Mehta(1902100), Harsh Mehta (1902098)** to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering” .**

### (Prof. Jayant Gadge )

Supervisor

### (Prof. Dr. Tanuja Sarode ) (Prof. Dr. G.T. Thampi )

Head of Department Principal

# Mini Project Approval

## The Mini Project entitled “Finger Tarp**”** by **Nishita Matlani (1902096), Isha Mehta (1902099), Shanvi Mehta(1902100), Harsh Mehta (1902098**) is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

**Examiners**

**1……………………………………**

(Internal Examiner Name & Sign)

### 2…………………………………………

(External Examiner name & Sign)

Date: Place:

# Contents

### Abstract ii

### Acknowledgments iii

### Introduction 1

* 1. Introduction
  2. Motivation
  3. Problem Statement & Objectives
  4. Organization of the Report

### Literature Survey 11

* 1. Survey of Existing System
  2. Limitation Existing system or research gap
  3. Mini Project Contribution

### Proposed System (e.g., New Approach of Data Summarization ) 18

* 1. Introduction
  2. Architecture/ Framework
  3. Algorithm and Process Design
  4. Details of Hardware & Software
  5. Experiment and Results
  6. Conclusion and Future work.

### References 32

# Acknowledgement

We would like to express our gratitude and thanks to **Prof. Supervisor Name** for her valuable guidance and help. We are indebted for her guidance and constant supervision as well as for providing necessary information regarding the project. We would like to express our greatest appreciation to our principal **Dr. G.T. Thampi** and head of the department **Dr. Tanuja Sarode** for their encouragement and tremendous support. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of the project.

Nishita Matlani

Isha Mehta

Shanvi Mehta

Harsh Mehta

# 

# Chapter 1

**Introduction**

This chapter explains the aim, objectives and scope of the proposed system.

## **Introduction**

Finger Tarp is an OpenCV project where a person can use their finger as a marker and write in the air. Writing in the air contributes immensely to the advancement in different fields one of which is digital art. Digital art refers to forms of expression and transmission of art form with the help of various technologies. Finger tarp is a system where in we use gesture recognition with the use of machine learning algorithm by using python programming. This creates natural interaction between man and the machine.

## **Motivation**

With the change in method of education all around the world and the concept of traditional training changed radically due to covid people found online education flexible and accessible. Now with advancement in education system the various different devices required to teach people can be replaced by this project. Pen and paper, chalk and board are examples of old writing methods. The primary goal of digital art is to create a hand gesture recognition system that can be used to write digitally. Many different methods of writing are used in digital art, such as using a keyboard, a touch-screen surface, a digital pen, a stylus, electronic hand gloves, and so on. However, with our system, we are utilizing hand gesture detection with a machine learning algorithm and Python programming to generate natural human-machine interaction.

**1.3 Problem Statement & Objectives**

Explaining topics can be a challenge without diagrams and working methods so this project makes it easier for professors to teach students without purchasing expensive devices and tablets.

People with hearing and listening problems can communicate with people who don’t understand sign language and they can freely exchange their thoughts and feelings without an actual translator between them.

**1.4 Organization of the Report**

This report consists of three chapters. The first chapter deals with introduction of the topic, problem statement, motivation behind the topic and objectives. The second chapter is the Literature Survey. It includes all the research work done related to this topic. All information related to study of existing systems as well as learning of new tools is mentioned in this chapter. The third chapter is about the proposed system which is used in this project. The block diagram, techniques used, hardware and software used screenshots of the project are presented in this chapter. All the documents related to development of this project are mentioned in References

# 

# Chapter 2

**Literature Survey**

This chapter explains the concepts used in this project, study of existing system and contribution of this project

**2.1 Survey of Existing System**

## A hands-free digital drawing canvas that utilizes a Raspberry Pi, a PiCamera, and OpenCV to recognize and map hand gestures onto a PiTFT screen. The user’s “brush” can be modified in size and color by using built-in buttons. The direction of the brush is controlled completely using open source OpenCV software and modified to map the pointer finger onto the screen using Pygame following a calibration screen to measure and record the color of the user’s hand. The idea for Air Canvas was a result of our interest in digital drawing and smart photo recognition software.

## The interface uses the parts listed below:

## Raspberry Pi Model B

## PiTFT Display

## Raspberry Pi Camera v2

## **2.2 Limitation of existing system**

## As mentioned above these systems are expensive and lack different things which are required in the project. This system overcomes all the aspects in the previous existing system and solves all the problems in a cost-effective manner Without any additional parts required

## **2.3 Mini Project Contribution**

This application has potential to change the traditional education system and can help save papers. It will also be a great help for people with especially abled people to communicate their emotions and feelings without the need of help from any third party. This can bring an excellent change in various aspects of life.

# Chapter 3

**Proposed System**

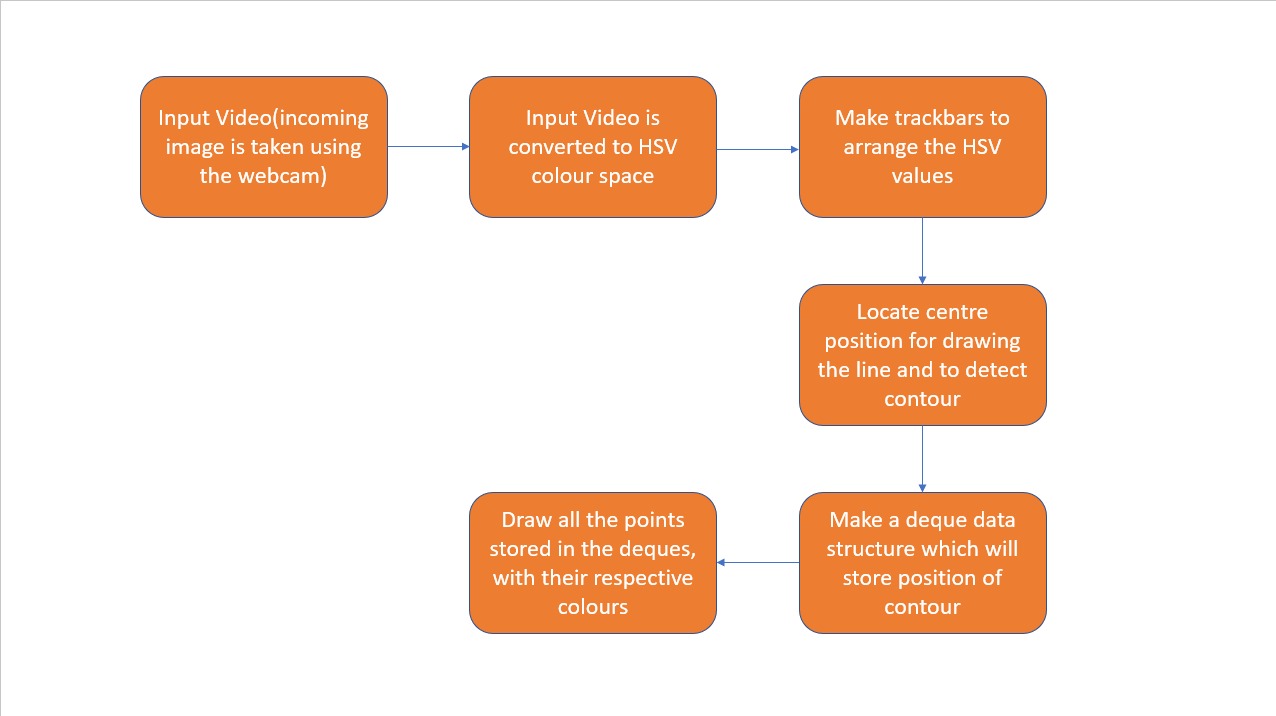
This chapter consists of detailed description about the methodology used, the hardware and software components, the tools used and also the screenshots of the project

**3.1 Introduction**

Air Canvas is a canvas that can be used to draw anything by just catching the action of a coloured marker with an **inbuild camera**. As a marker, a coloured object at the tip of the finger is used.

This project will be built with **OpenCV's** computer vision techniques. The ideal language is Python because of its extensive libraries and simple syntax, but after you learn the principles, you can use any OpenCV supported language.

## **3.2 Architecture**



## The above diagram represents the entire working of the model in detail.

## **3.3 Algorithm and Process Design**

1. Formulating the Problem statement

Air Canvas' main purpose is to transfer the user's pointer finger coordinates to the screen, where colored circles are formed and joined to resemble a simple brush stroke.

That is a computer vision project, which allows you to create on a screen just by waving your finger with a colorful point or a basic-colored cap. For these computer vision tasks, it was OpenCV that came to the rescue. The suggested technique allows for natural human-system interaction by eliminating the need for a keyboard, stylus, pen, or glove to input characters.

2. Understanding the framework and requirements

Researched and gathered information about the requirements to make an OpenCV project. This helped in making the application as user friendly as possible.

3. Identifying tools/technology to be used

Browsed the web regarding what previous knowledge was required to create an OpenCV project and what tools/technology were required to implement the working of Finger Tarp, a digital art project

4. Finalizing the features to be included

Decided which features were best to make Finger Tarp work as efficiently as possible and also the time required to implement it .

5. Development

* Color Tracking

Color Tracking requires knowledge of the HSV color space (Hue, Saturation, and Value). In addition, the small colored item at the tip of the finger is being tracked.

The incoming picture from the camera must be transformed to the HSV color space in order to detect the colored item near the tip of the finger. The input picture is converted to the HSV color space, which is ideal for color tracking.

In an application where you want to track a user’s hand movement, skin color histogram will be very useful. This histogram is then used to subtracts the background from an image, only leaving parts of the image that contain skin tone.

A much simpler method to detect skin would be to find pixels that are in a certain RGB or HSV range.

Trackbars Are designed to align the HSV values to the desired color range of the colourful item we've put on our finger Various HUE and other product lines of different colours.

We'll obtain the Realtime value from the trackbars and construct a range once they're set up. This range is a NumPy structure that is given to the cv2.inrange( ). The Mask on the colored object is returned by this function. This Mask is a black-and-white picture with white pixels in the desired color's location.

* **Contour Detection**

Forming a circle over a colored item at the tip of the finger. Contour Detection

Is the term for it. Prepare the canvas frame by placing the appropriate ink buttons on top of it.For determining the colored marker mask, adjust the trackbar values. Locate its center position for drawing the Line.

We'll create a Python deque, the deque will save the location of the contour on each frame, and we'll utilize these points to create a line with OpenCV drawing algorithms.

We'll now use the location of the contour to determine whether we want to click a button or draw on the page. We've positioned some of the buttons on the top of Canvas such that when the cursor enters their region, they'll be triggered. On the canvas, we have four buttons created using OpenCV.

* Clear : clears the screen by emptying the deques.
* Red : Changes to red colour using colour array.
* Green : Changes to green colour using colour array.
* Yellow : Changes to yellow colour using colour array.
* Blue : Changes the marker to blue colour using colour array.
* **Frame Processing**

For the air canvas effect, track the fingertip and draw points at each place. That's how **Frame Processing** works. We will draw all the points on the positions stored in the deques, with respective colors

* **Algorithmic Optimization**

Fixing minor details in the code in order for the application to run properly. **Algorithmic Optimization** is a term that refers to the process of optimizing

6. Testing

While testing the application Any colored pointer could be tracked. The user can draw in four distinct colors and easily switch between them. A single spot at the top of the screen may be used to rub the board. Once the software is started, there is no need to touch the computer.

7. Evaluation

Received feedback regarding the project/application from a few family members & classmates and made the required changes.

**3.4 Details of Hardware & Software**

Hardware Requirements

Processor – Minimum 1Ghz

Ethernet Connection – LAN or Wi-Fi

Hard Drive – Minimum 32GB

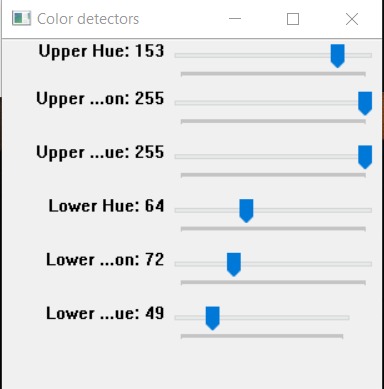
Memory (RAM) – Minimum 4GB

Software Requirements

Backend – Python (Anaconda is preferred for the required modules)

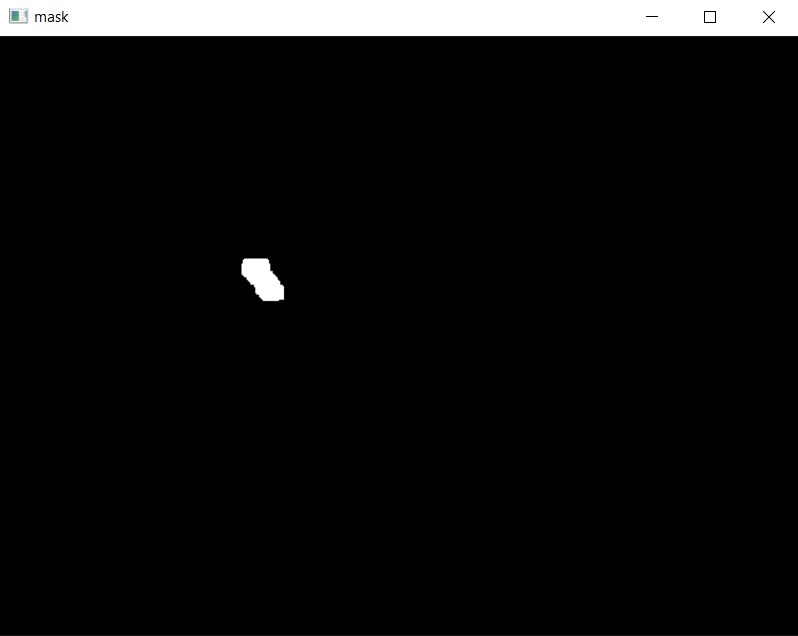
OpenCV version 4.1

**3.5 Results**

****

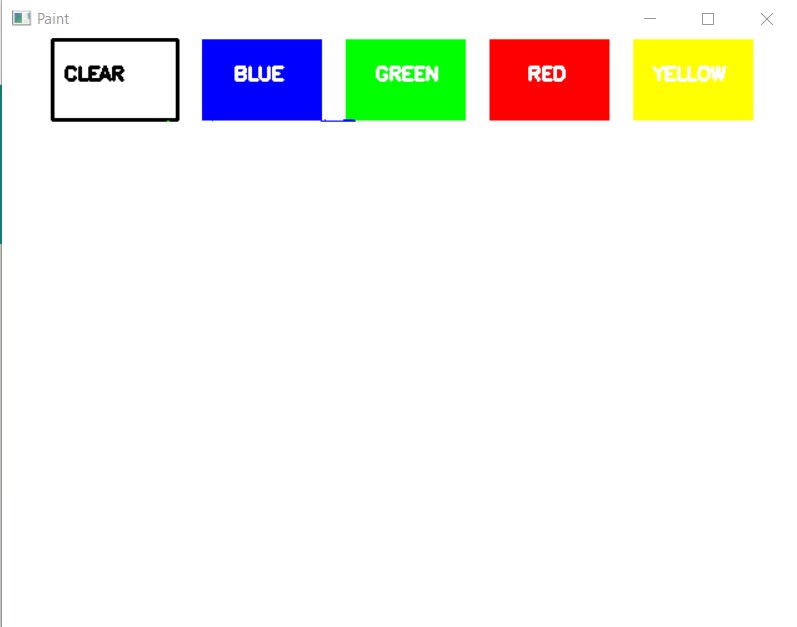
**Color Detector Screen**

The colour of the bead we are going to use in our project here we have set the saturation value for that bead

****

**Mask Screen**

The bead is detected and that froms a mask

****

**Paint Screen**

**3.6 Conclusion and Future Work**

Created a good and satisfactory project allowing users to freely write on the computer screen without the hassle of expensive devices such as stylus, electronic gloves etc.

It also has a user-friendly interface making it easier for abled people to use. It will help them communicate better. It will also help in saving paper which in turn saves the environment.

To enhance hand gesture tracking, we would have to delve more into OpenCV. There are many different methods of contour analysis, but in this particular algorithm, it may be worthwhile to take a look at the color histogram used to create the contours in question

**References**

[1]<https://dev.to/amarlearning/finger-detection-and-tracking-using-opencv-and-python-586m>

[2]https://docs.opencv.org/4.5.3/ [3]OpenCVcontours: <https://docs.opencv.org/3.4.2/d4/d73/tutorial_py_contours_begin.html>