# PROFESSIONAL TRAINING REPORT

# at

**Sathyabama Institute of Science and Technology**

**(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award

of Bachelor of Engineering Degree in Computer Science and Engineering

By

## RENDLA SAINATH REDDY

**(Reg.No: 40111054)**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**SCHOOL OF COMPUTING**

**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY JEPPIAAR NAGAR, RAJIV GANDHI SALAI,**

**CHENNAI – 600119, TAMILNADU**

**APRIL 2023**

**SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY**

### (DEEMED TO BE UNIVERSITY)

**Accredited with Grade “A” by NAAC**

**(Established under Section 3 of UGC Act, 1956)**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI– 600119**

[**www.sathyabamauniversity.ac.in**](http://www.sathyabamauniversity.ac.in/)

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of **RENDLA SAINATH REDDY (40111054)** who carried out the project entitled “**COLOUR RECOGNITION** **USING PYTHON”** under my supervision from January 2023 to April 2023.

## Internal Guide

## Ms. LAKSHMI PRIYA S,M.E,(Ph.D)

**Head of the Department**

**Dr.L.Lakshmanan M.E., Ph.D.,**

## Submitted for Viva voce Examination held on

**Internal Examiner External Examiner**

**DECLARATION**

I **RENDLA SAINATH REDDY** hereby declare that the Project Report entitled **COLOUR RECOGNITION** Done by me under the guidance of **Ms. LAKSHMI PRIYA S,M.E,(Ph.D)** at **HCL INTERNSHIP** (Company name and address) is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

## DATE:

**PLACE: SIGNATURE OF THE CANDIDATE**

**ACKNOWLEDGEMENT**

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D.**, **Dean**, School of Computing, **Dr.S.Vigneshwari M.E., Ph.D., and Dr.L.Lakshmanan M.E., Ph.D.,** Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms. LAKSHMI PRIYA S,M.E,(Ph.D)** for her valuable guidance, suggestions and constanten couragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

# ABSTRACT

A Color detection technique is a program that gives the color of what user asks for as output. In this method the color codes already provided in the program are compared it. the image whose colors we want to know. Let us first began with. We know of a disability in humans very commonly found “color blindness” it is very common in humans but mostly people manage with it, as if you don’t go deeply about the colors it won’t cause any problem to you in general activities.

|  |  |  |
| --- | --- | --- |
|  | **TABLE OF CONTENTS** |  |
| **CHAPTER No.** | **TITLE** | **PAGE No** |
| **1.** | **INTRODUCTION** | **7-15** |
| 1.1About python | 7 |
| 1.2 what is color detection | 8 |
| 1.3 What is color Blindness | 8 |
| 1.4How does Color Detection  System Works | 9 |
| 1.5System architecture | 11 |
| 1.6Definition of terms | 14 |
| 1.7 Objective of the study | 15 |
| 1.8 Significance of the study | 15 |
| **2.** | **AIM AND SCOPE OF THE PRESENT INVESTIGATION** | **16-17** |
| 2.1 Aim | 16 |
| 2.2 Scope | 17 |
| **3.** | **ALGORTHIMS USED AND IMPLEMENTATION** | **18-22** |
| 3.1 Algorithms used | 18 |
| 3.2 Implementation | 21 |
| **4.** | **RESULTS & DISCUSSION** | **23-26** |
| 4.1 Results | 23 |
| 4.2 Conclusion | 25 |
| **5.** | **REFERENCES** | **27** |
| **6.** | **SOURCE CODE** | **28-36** |

**CHAPTER 1**

**INTRODUCTION**

* 1. **About Python**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

* 1. **what is color detection.**

Before going into the speculations of the project it is important to know the definition of color detection. It is simply the process of identifying the name of any color. It is obvious that humans performs this action naturally and do not put any effort in doing so. While it is not the case for computers. Human eyes and brain work in coordination in order to translate light into color. Light receptors that are present in eyes transmit the signal to the brain which in turn recognizes the color. There is no exaggeration in saying that humans have mapped certain lights with their color names since childhood. The same strategy is useful in detecting color names in this project. Three different colors Red, Green and Blue are being tracked by utilising the fundamentals of computer vision. After successful compilation when we execute the code a window redirects the image displayed on it whose path is given as an argument.

Color is a trait of wavelengths of which light consists of. The human eye is capable of perceiving color of light between wavelengths of 750 and 400 nm. All wavelengths together are called the electromagnetic spectrum of which the human can only see a small portion.

Color is generated when objects are illuminated by electromagnetic radiation in a wavelength area between 380 nm and 780 nm. This area is perceivable for the human eye. The radiation with a wavelength of less than 350 nm is called ultraviolet radiation. Radiation with a wavelength of more than 750 nm is called infrared radiation. These ends of the spectrum are invisible to the human eye. Color can also be generated by primary light sources that radiate light in the (for the human) visible range between the 380 and 780 nm.

* 1. **What is color Blindness.**

Color blindness occurs when you are unable to see colors in a normal way. It is also known as color deficiency. Color blindness often happens when someone cannot distinguish between certain colors. This usually happens between greens and reds, and occasionally blues.

Color blindness can happen when one or more of the color cone cells are absent, not working, or detect a different color than normal. Severe color blindness occurs when all three cone cells are absent. Mild color blindness happens when all three cone cells are present but one cone cell does not work right. It detects a different color than normal.

The inability of the human eyes in which they won’t be able to perceive red, yellow, blue or green colours, is called the colour blindness. Colour blindness is not any form of blindness but it is the inability or deficiency of the way a person sees colours. Normally, any person who is colour blindness will have trouble seeing one or two colours and they will perceive them in a different way in comparison with the person with normal colour vision. For example, if a person has blue-yellow colour blindness, then they may see these colors red and green.

When any person watches an object, it’s image is being projected on the screen which is called retina and is located at the back of the eyes. Retina is a neural membrane line which is made up of two types of photoreceptor cells which are known as rods and cones. These are responsible for perceiving objects which are in the dark or at night and won't be able to perceive color. Approximately, there are 120 million rods present in each retina. Cones are responsible for perceiving the colors in the presence of light.

**1.4 How Does Color Detection System Works.**

Color detection is the act to detect and identify the color in the image/video asked or clicked on.

It captures, analyzes, and compares the R,G,B value and gives the corresponding color.

1. The Color detection process is an important step because it detects and saves the colors of images and videos.

2. The image capture process the converts analogue (color) information into a set of digital data (R, G, B values).

3. The color recognition It can be caused because of chronic diseases like diabetes, etc.

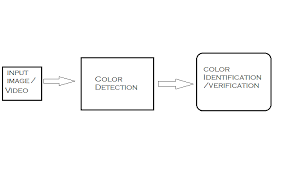
**CAUSES OF COLOR BLINDNESS**

Color blindness can be caused when the medications are being used for a very long time specially those help in the treatment of the nervous disorders. It can also be caused if any accident has happened and which has caused any damage to the eyes or has restricted the supply of the blood to the eyes. Color blindness can be caused because of the age as it is a part of physical changes.

It is mostly caused due to inheritance. It usually comes from parents. The chromosome carries the gene which is responsible for color blindness. Due to this, men get more affected.

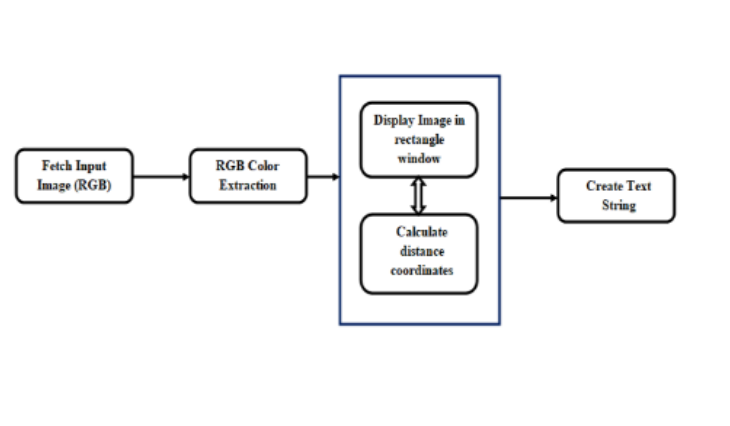
But some severe accidents, illness, and medication can also cause color blindness. The healthy eye retina of humans consists of rod and cone cells. So if these cells fail to perform then it leads to color blindness. If a person’s eyes have no cones or only one cone, it causes monochromacy.

Dichromacy is caused if one type of cone is missing. When all cones are present but not aligned, it causes anomalous trichromacy in these cases. n process compares and confirms whether the two colors same or not



**FIG 1.1 COLOR RECOGNITION**

**1.5 SYSTEM ARCHITECTURE**



**FIG 1.2**

**SYSTEM ARCHITECTURE DIAGRAM FOR COLOR RECOGNITION**

In the above architecture shows the capability for the project. It consists of a

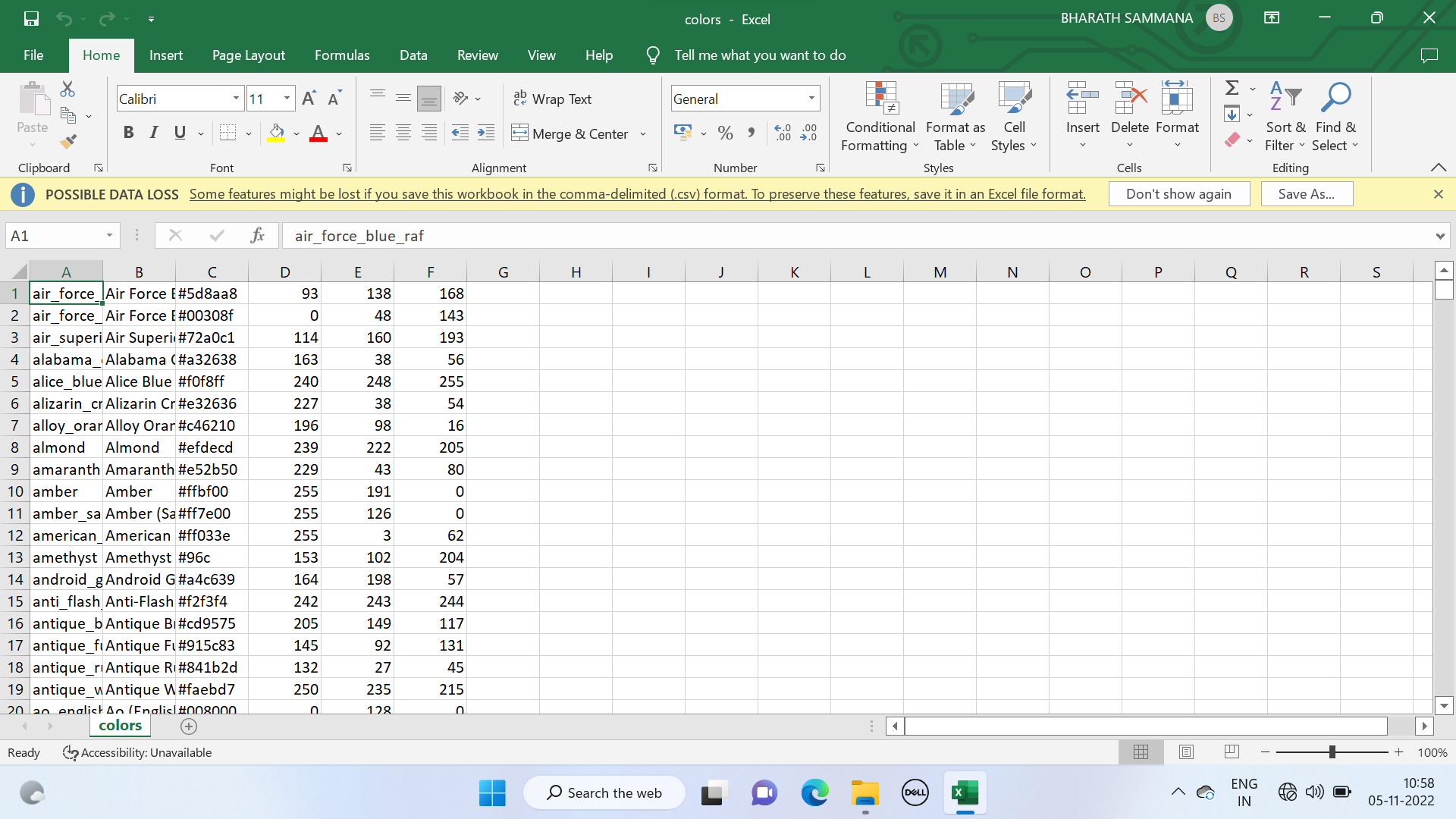
well defined sequence diagram that is abstracted from the source code. It leverages the rich capabilities of the technology such as OpenCV library in the python.

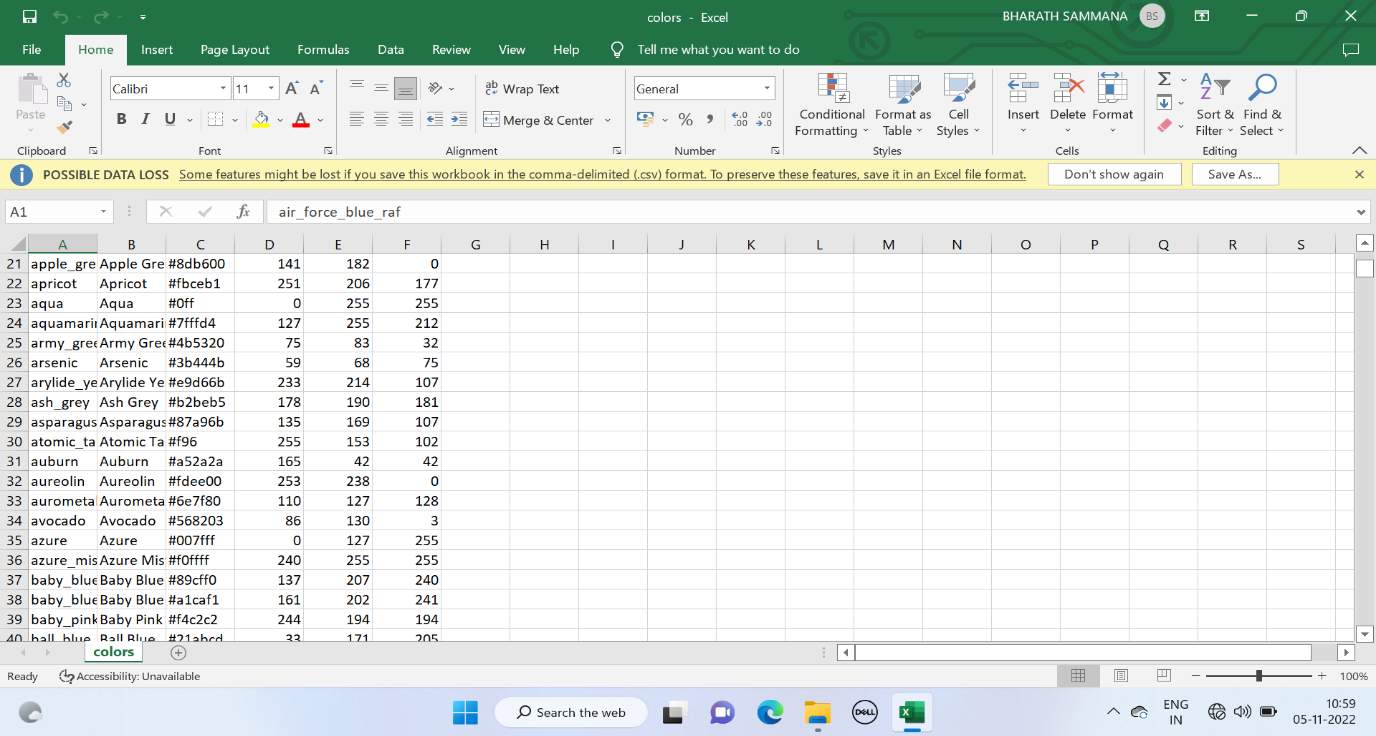
The above architecture makes the process more efficient based on principles and properties related to each other. As we know the red, green and blue are the primary colors that can be mixed to produce different colors.

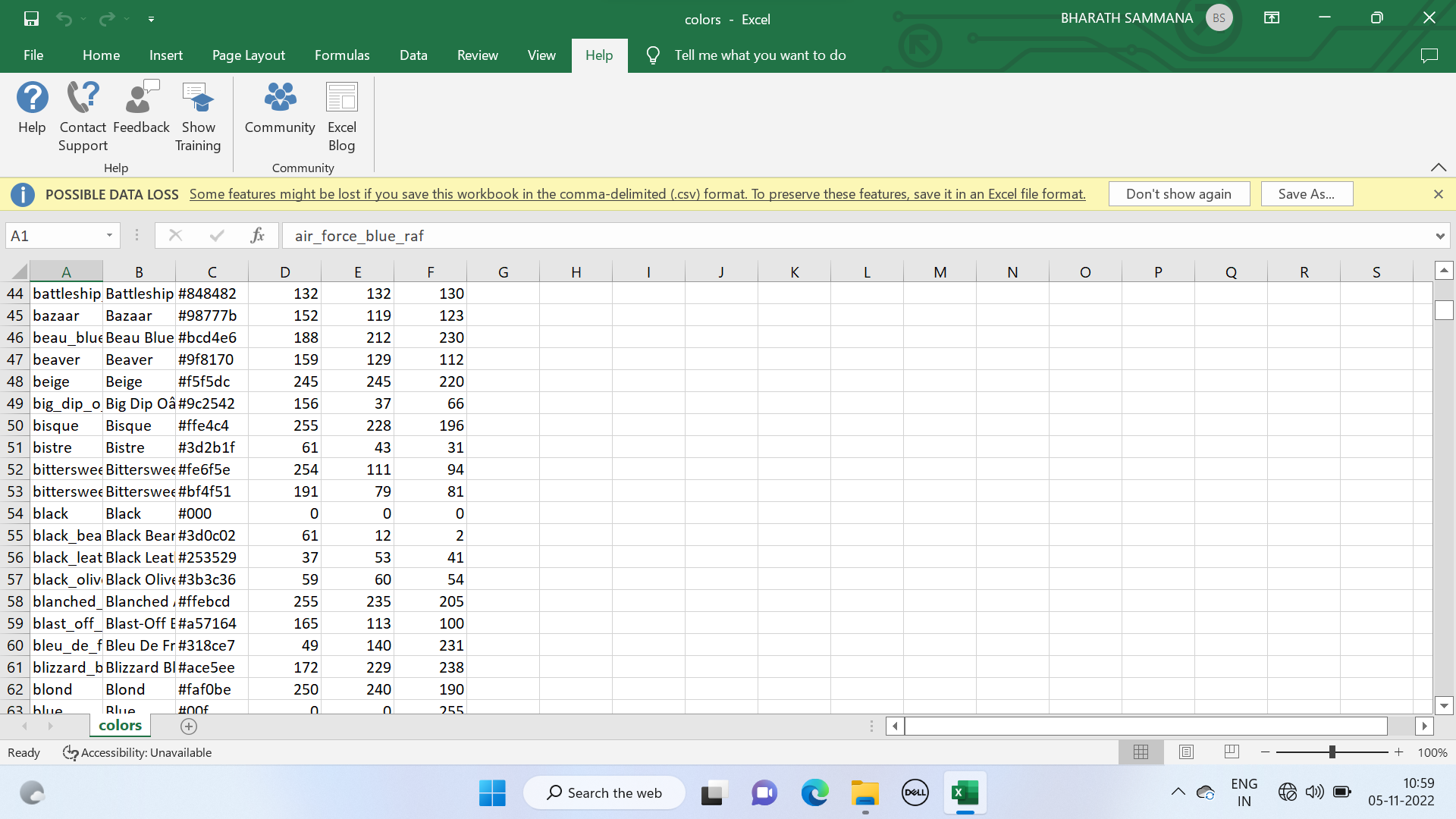
The present color detection project takes the path of an image as an input and looks for the composition of three different color red, green and blue in an image.

Above architecture makes the process more efficient based on principles and properties related to each other. As we know the red, green and blue are the primary colors that can be mixed to produce different colors.

The present color detection project takes the path of an image as an input and looks for the composition of three different color red, green and blue in an image.



****

****

**1.6 Definition of terms**

Gray Conversion:

It is the process of conversion from color images into gray scale image. Color image contains 24 bits per pixel: it is reduced into 8 bits per pixel. Most commonly gray levels represent the interval number of quantization in gray scale image processing. At present, the most commonly used storage method is 8-bit storage. There are 256 gray levels in an 8 bit gray scale image, and the intensity of each pixel can have from 0 to 255. Subtraction: The RGB image contains 24 bits, each of three colour shaving 8 bits per pixel. At parallel RGB is splitting into each 8 bit colours.

The colour subtraction is the process of subtracting the colour values between the two colours. Here each three colours are subtracted with the gray image which converted from the original RGB image. Binary Conversion: Binary conversion is the process of converting any kind of image into a binary image. Basically binary image two bits image, it contains only 0 and 1.Here I will be represented as white and 0 will be represented as black. Hence it's called as black and white image. The purpose of conversion is to count the white and black pixels in the image. Each separated colour is converted as white and other colours are converted as black.

Multiplication Image:

multiplication is the process of multiplication of pixel values between two or more images: here this process is used to multiply binary images with splitted color images. At the end of this process we can get each primary colors that having above 300 pixels. The area contains below 300 pixels does not considered as color. After the multiplication this process having three segmented colors that having 300 pixels. By combining these three colors we can get the segmented color image that contains only primary colors.

Binary Conversion:

Binary conversion is the process of converting any kind of image into a binary image. Basically binary image two bits image, it contains only 0 and 1.Here I will be represented as white and 0 will be represented as black. Hence it's called as black and white image. The purpose of conversion is to count the white and black pixels in the image. Each separated colouris converted as white and other colours are converted as black.

**1.7** **OBJECTIVE OF THE STUDY**

In this color detection Python project, the objective of this study is to build an application through which you can automatically get the name of the color by clicking on them**.**

**1.8 SIGNIFICANCE OF THE STUDY**

We will be working with colors and you will get to learn about many concepts throughout this project. Colour detection is necessary to recognize objects, it is also used as a tool in various image editing and drawing apps..7 object of study

**CHAPTER 2**

**AIM AND SCOPE**

**AIM:**

The aim of the project is Colour Recognition in this we are going to build an application through which you can automatically get the name of the color by clicking on them. So for this, we will have a data file that contains the color name and its values. Then we will calculate the distance from each color and find the shortest one and it also shows the RGB values of the particular colour we choose

**SCOPE :**

Color detection can easily be implemented and helps the user to get the colors easily. Colors are as true as the sun or the moon and is always present in the environment, but unlike normal people, people suffering from color blindness can't easily identify the colors. We are seeing an increase in people suffering from this and we know that it can be very frustrating to not able to see the colors what least can be done is to tell them the real colors name. ally based on clusters, taken at the time of arrest, before the judge includes the possibility of conviction or innocence. Mugshot photos are often not far from the details, even though the detainee has never been charged.

**1.EXISTING SYSTEM** :

In the existing system they have gone through with the opencv but while extraction of the colors they got the wrong outputs. There is no exact color representation of colors with accuracy. existing system is based on the color joint probability function, in this it will look for the centroid of the colors and color edge co-occurrence histogram, so that the accuracy of this system will less.

**2.PROPOSED SYSTEM** :

In the proposed system, we are introducing the CV database and according to it the number of shades that can be identified using 865 color names along with their RGB and hex values. Whenever the cursor clicks the image, it automatically shows the RGB shades color values.

[This paper presents a hardware architecture for increased performance of color classification. In our architecture, color classification, based on an AdaBoost algorithm, identifies a pixel](https://medium.com/programming-fever/color-detection-using-opencv-python-6eec8dcde8c7" \t "_blank)

**CHAPTER 3**

**ALGORTHIMS USED AND IMPLEMENTATION**

* 1. **ALGORTHIMS USED:**

**a)OpenCV:**

The OpenCV is an opensource library of computer vision and image processing etc. It plays an important role in real time image operation, and it is an important part of today computer systems. By using this software the user can process image, detect objects and this library of OpenCV is gradually evolving because of its ability to perform more complex tasks in processing images etc in a consistent manner. This library has been applied extensively in companies, public bodies (like Government bodies), well established software companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda make extensive use of OpenCV. Color detection is necessary to recognize objects, it is also used as a tool in various image editing and drawing apps. I write a simple Python code to detect the color in the image using OpenCV and pandas

**b) Pandas**:

Pandas is Python Package which stands for Python and data analysis.

This library helps in data manipulation and analysis.it is built upon top of another package called Pandas is an open-source library that is made working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

**History:**

Pandas were initially developed by Wes McKinney in 2008 while he was working at AQR Capital Management. He convinced the AQR to allow him to open source the Pandas. Another AQR employee, Chang She, joined as the second major contributor to the library in 2012. Over time many versions of pandas have been released. The latest version of the pandas is 1.5.0, released on Sep 19, 2022

### Advantages

* Fast and efficient for manipulating and analyzing data.
* Data from different file objects can be loaded.
* Easy handling of missing data (represented as NaN) in floating point as well as non-floating point data
* Size mutability: columns can be inserted and deleted from DataFrame and higher dimensional objects
* Data set merging and joining.
* Flexible reshaping and pivoting of data sets
* Provides time-series functionality.
* Powerful group by functionality for performing split-apply-combine operations on data sets.

**c) Numpy**:

it is a library of python which adds support for multi - dimensional arrayand matrices along with a large collection of mathematical functions to operate on the array a general-purpose array-processing package. It provides a high performancemultidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

**d) Image processing:**

Image Processing technique is used to perform some certain operations on an image, in order to get an enhanced image as an output or to extract some useful information from the image.

We all would have cropped our photos, rotated them, added some filters, etc. Have you ever thought of doing these by using your code? In this article, you will be able to get insights into the concept of image processing using Python. We will see different libraries Python provides for this purpose. Then we will discuss in detail the libraries numpy, scipy, OpenCV, and PIL.

Before discussing processing an image, let us know what does an image means?  
Image is a 2D array or a matrix containing the pixel values arranged in rows and columns. Think of it as a function F(x,y) in a coordinate system holding the value of the pixel at point (x,y).

For a grayscale, the pixel values lie in the range of (0,255). And a color image has three channels representing the RGB values at each pixel (x,y), each varying from 0 to 255. It now becomes a 3-dimensional array.

Image processing, as the name suggests, is a method of doing some operation(s) on the image. You would have also heard of another term called ‘Computer Vision. The difference is that in image processing we take an input image, do required changes, and output the resulting image. Whereas, in Computer vision, we look for some features or any other information related to the input image.

Different actions are performed on the images for different applications which include cropping, flipping, rotation, segmentation, etc. Python provides functions for all these methods, using which we can set parameters that suit our needs. There are different modules in Python which contain image processing tools.

**3.2 IMPLEMANTATION**

**Image Capture :**

The first step is to fetch a high-quality image with resolution. To load an image from a file we useCv2.imread(). Image should be in working directory or full path of the image should be given. Img=cv2.imread(img path)

In this article, we will discuss how to capture an image from the webcam using Python.

We will use opencv and pygame libraries. Both libraries include various methods and functions to capture an image and video also. By using, these vast libraries we need to write only 4 to 5 lines of code to capture an image

**Extraction of RGB Colors:**

In this phase, the 3 layered colors are extracted from the input image. All the color images on screens such as televisions, computer, monitors, laptops and mobile screens are produced by the combination of Red ,Green and Blue light.

Each primary color takes an intensive value 0 (lowest) to 255 (highest). When mixing 3 primary colors at different intensity levels a variety of colors are produced. For Example: If the intensity value of the primary colors is 0, this linear combination corresponds to black. If the intensity value of the primary colors is 1, this linear combination corresponds to white.

Index=[ "color", "color name", "hex", "R", "G", "B"]

Any image consists of pixels, each pixel represents a dot in an image. A pixel contains three values and each value ranges between **0 to 255**, representing the amount of **red**, **green** and**blue** components. The combination of these forms an actual color of the pixel. To find the dominant colors, the concept of the k-means clustering is used. One important use of k-means clustering is to segment satellite images to identify surface features.

**Calculate minimum distance from coordinates:**

The minimum distance is calculated by considering moving towards the origin point from all colors to get the most matching color.

The pandas library serves as an important utility to perform various operations on

Comma -seperated values like pd.read\_csv() reads the csv file and loads it into the pandas data frame.

D= abs(R-int(csv.loc[i ,"R"])) + abs (G-int (csv.loc[i ,"G"])) + abs (B- int (csv.loc [i ,"B"])).

**Image Display with Shades of Color:**

The rectangle window is used to display the image with shades of color. After the double-click is triggered, the RGB values and color name is updated. To display an image Cv2.imshow () method is used. By using cv2.rectangle and cv2.putText () functions, the color name and its intensity level can be obtained.

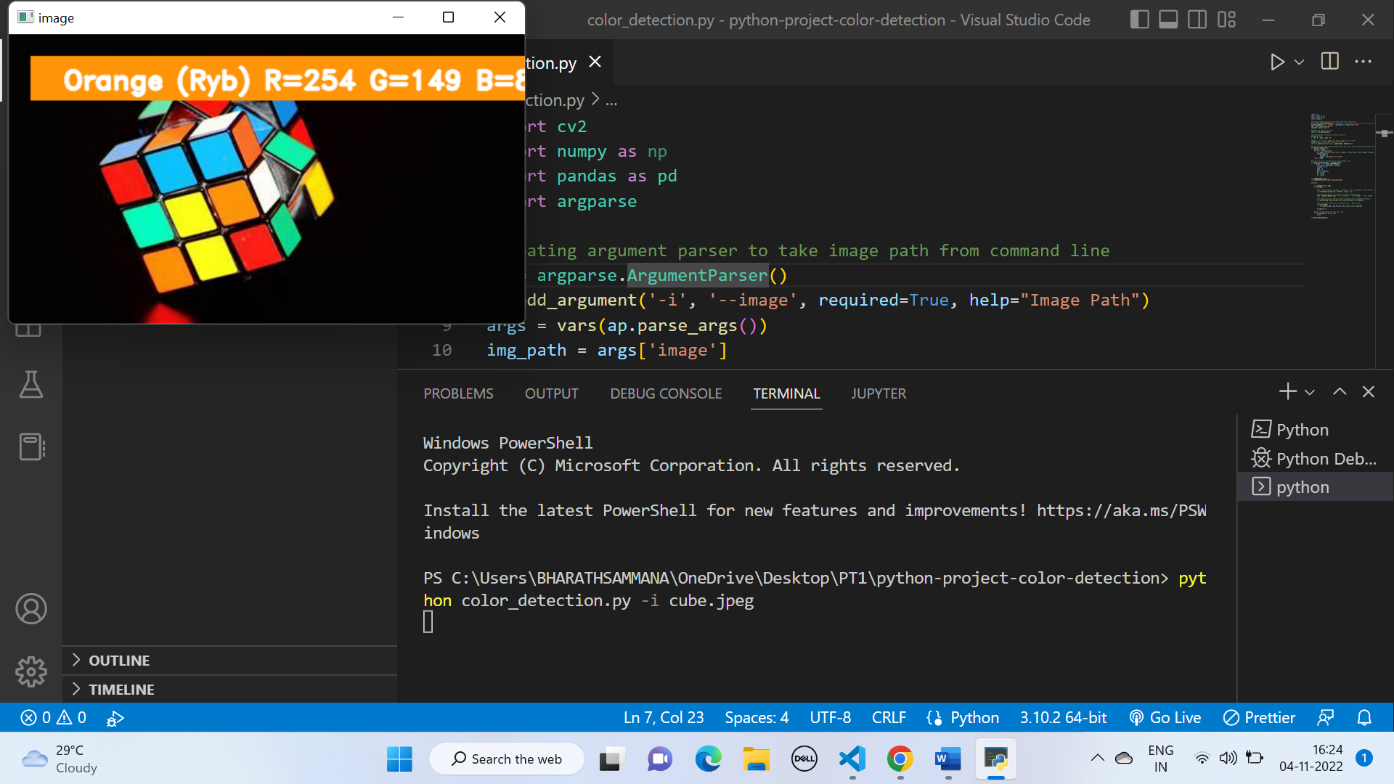
text=getColorName(r,g,b) + 'R='+str(r) + 'G='+str(g) + B=' +str(b).I have files in two type: A: contains 1206 lines of coordinates (xyz) - a protein chain B: contains 114 lines of coordinates (xyz) - a bunch of moleculeI would like to do the followings: For each line of A calculate distance from each line of B. So I get 114 distance value for each line of A. But I don't need all of them, just the shortest for each line of A. So the desired output: A file with 1206 lines, each line contains one value: the shortest distance. Important to keep the original order of file A.

**CHAPTER 4**

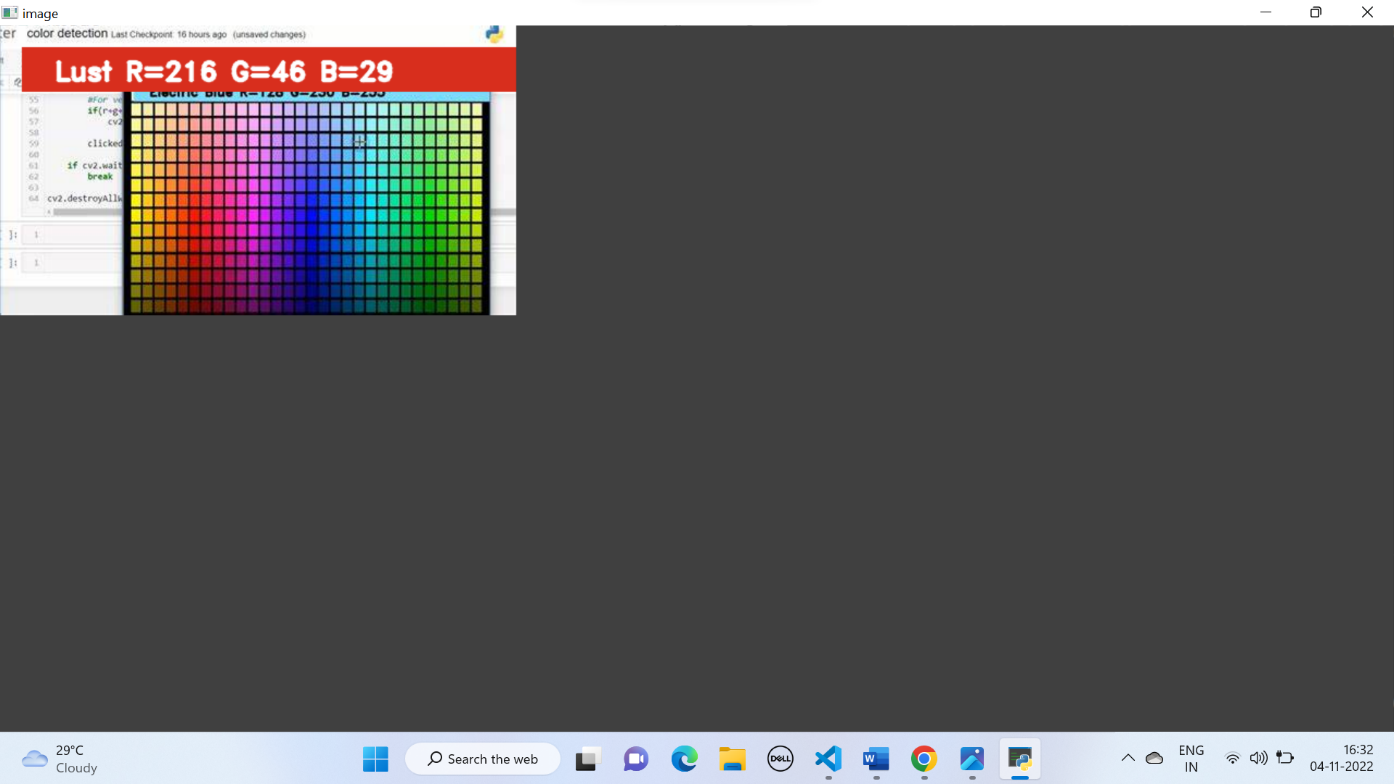
**RESULT AND DISCUSSION**

**4.1 Result:**

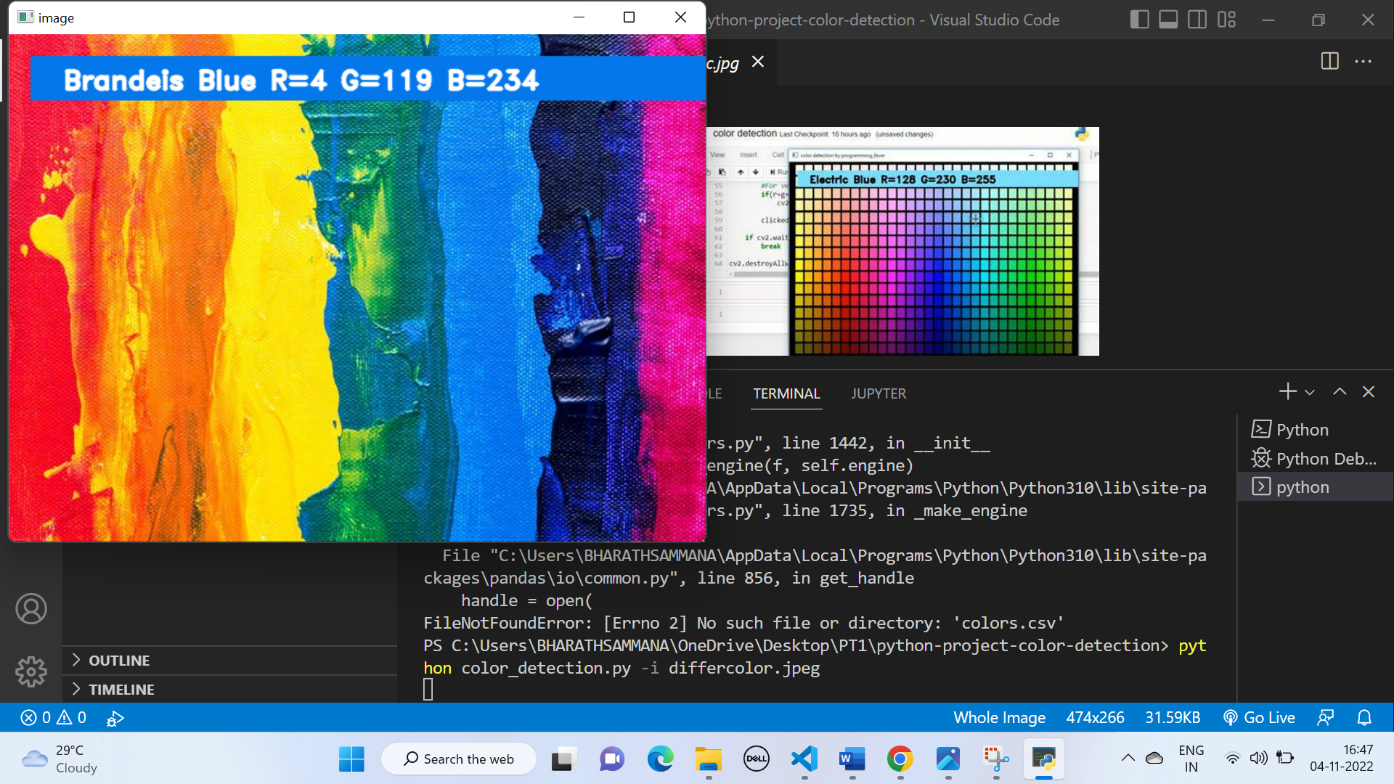
The process of detecting the color is successfully completed using the color.csv file and when we double click on the picture to know the color and it shows with RGB values



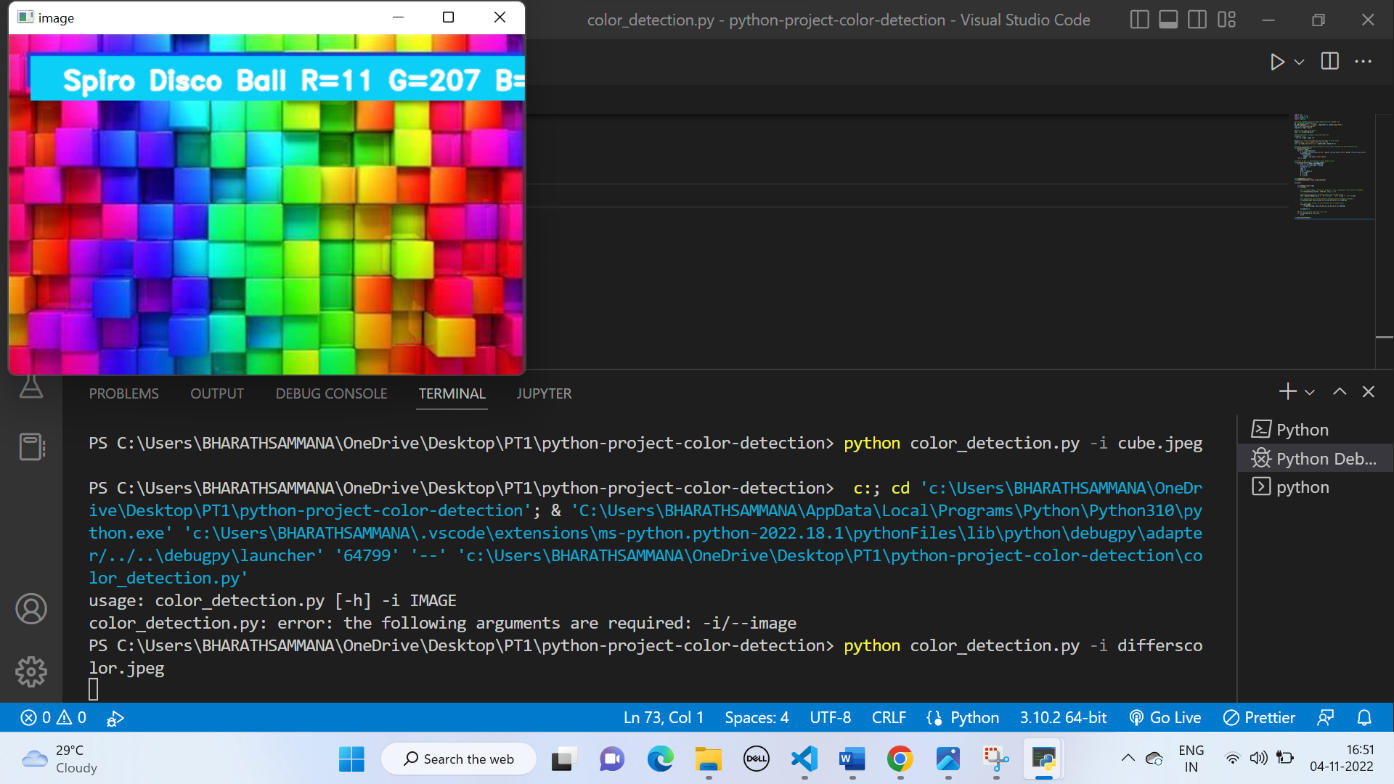
**FIG 4.1.1 cube**



**FIG 4.1.2 color pic**



**FIG 4.1.3 various colors**



**FIG 4.1.4 Differcolor**

**4.2 CONCLUSION**

In this paper we defined to get the required color field from an RGB image. In this various steps are implemented using openCV platform. The main positive point of this method is its color differentiation of a mono color. In the future scope, the detection of the edge detection techniques has different other applications like facial detection color conversion for grey scale image etc. That can also be implemented

Color detection technology has come a long way and has a long way to go. When we see selfdrive cars running on roads by themselves following the traffic rules. Today, the machines are ready to for it. Tesla is a frontrunner in this technology. However, next-generation color detection programs will have more upgradations. The apps in smart environments - where computers and equipment are similar to assistant assistants

To achieve this goal computers must be able to reliably identify nearby things and their basic properties like size shape and color(we can’t forget that) in a manner that is naturally consistent within the normal human pattern. They do not require special interactions and should be in line with people's understanding of when recognition goes. This suggests that future intelligent environments should use the same methods as humans, and have the same limitations. These goals are now achievable.

**CHAPTER 5**

**REFERENCES:**

1. OpenCV– medium.com

2. Color detection \_–Medium.com

3. Weiming Hu, Xue Zhou, “Active Counter Based Visual Tracking by Integrating

Colors, Shapes and Motions”, IEEE TRANSACTIONS ON IMAGE PROCESSING,

VOL. 22, NO. 5, MAY 2013

4. Color Blindness – Wikipedia

5.Weiming Hu, Xue Zhou, “Active Counter-Based Visual Tracking by Integrating Colors, Shapes and Motions”, IEEE TRANSACTIONS ON Volume 7, Issue 1, January-February-2021 | http://ijsrcseit.com IMAGE PROCESSING, VOL. 22, NO. 5, MAY 2013

6. G.M. Snoek, “Evaluating Color Descriptors for Object and Scene Recognition”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 32, NO. 9, SEPTEMBER 2010

7. Claudia Nieuwenhuis, “Spatially Varying Color Distributions for Interactive Multi Label Segmentation”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL 35, NO. 5, MAY 2013

8. Kok -Meng Lee, “Effects of Classification Methods on Color-Based Feature Detection with Food Processing Applications”, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, VOL. 4, NO. 1, JANUARY 2007

9. J. Van de Weijer, “Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields”, IEEE TRANS PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 23, N0. 9, PP. 1035- 1042, APRIL 2001

10. Alexander Toet, “Multisource Information Fusion Architectures Algorithms and Applications” SPIE 6947 BELLINGHAM WA USA THE INTERNATIONAL SOCIETY FOR OPTICAL ENGINEERING, 1 12, 2008

# CHAPTER 6

**SOURCE CODE**

import cv2 # image loading and displaying

import pandas as pd # quick calculation

import argparse # taking arguments

import os

import numpy as np

#Creating argument parser to take image path from command line

#ap = argparse.ArgumentParser()

#ap.add\_argument("-i", "--image", required=True, help="Image Path")

#args = vars(ap.parse\_args())

#img\_path = args["image"]

def imageto():

img\_path=input("Enter the image name: ")

# Reading the image with opencv

img = cv2.imread(os.getcwd()+"\\"+img\_path)

#height, width, \_ = img.shape

# square root of sums of squares of the sides is the actual max

# (Pythagoras Theorem) but that requires importing the math module

max\_distance = 1000

print("Press Esc to exit the program! :)")

# declaring global variables (are used later on)

clicked = False

r = g = b = xpos = ypos = 0

# Reading csv file with pandas and giving names to each column

index = ["color", "color\_name", "hex", "R", "G", "B"]

csv = pd.read\_csv("colors.csv", names=index, header=None)

# function to calculate minimum distance from all colors and get the most matching color

def getColorName(R, G, B):

minimum = max\_distance

for i in range(len(csv)):

d = (

abs(R - int(csv.loc[i, "R"]))

+ abs(G - int(csv.loc[i, "G"]))

+ abs(B - int(csv.loc[i, "B"]))

)

if d <= minimum:

minimum = d

cname = csv.loc[i, "color\_name"]

return cname

# function to get x,y coordinates of mouse double click

def draw\_function(event, x, y, flags, param):

if event == cv2.EVENT\_LBUTTONDOWN:

global b, g, r, xpos, ypos, clicked

clicked = True

xpos = x

ypos = y

b, g, r = img[y, x]

b = int(b)

g = int(g)

r = int(r)

# driver code

cv2.namedWindow("Photo")

cv2.setMouseCallback("Photo", draw\_function)

while True:

cv2.imshow("Photo", img)

if clicked:

cv2.rectangle(img, (20, 20), (750, 60), (b, g, r), -1)

colorName = getColorName(r, g, b) + " R=" + str(r) + " G=" + str(g) + " B=" + str(b)

cv2.putText(img, colorName, (50, 50), 2, 0.8, (255, 255, 255), 2, cv2.LINE\_AA)

if r + g + b >= 600:

cv2.putText(img, colorName, (50, 50), 2, 0.8, (0, 0, 0), 2, cv2.LINE\_AA)

clicked = False

# Exit Program if esc key is pressed

if cv2.waitKey(20) & 0xFF == 27:

break

cv2.destroyAllWindows()

def camto():

webcam = cv2.VideoCapture(0)

# Start a while loop

while(1):

\_, imageFrame = webcam.read()

# Convert the imageFrame in

# BGR(RGB color space) to

# HSV(hue-saturation-value)

# color space

hsvFrame = cv2.cvtColor(imageFrame, cv2.COLOR\_BGR2HSV)

# Set range for red color and

# define mask

red\_lower = np.array([136, 87, 111], np.uint8)

red\_upper = np.array([180, 255, 255], np.uint8)

red\_mask = cv2.inRange(hsvFrame, red\_lower, red\_upper)

# Set range for green color and

# define mask

green\_lower = np.array([25, 52, 72], np.uint8)

green\_upper = np.array([102, 255, 255], np.uint8)

green\_mask = cv2.inRange(hsvFrame, green\_lower, green\_upper)

# Set range for blue color and

# define mask

blue\_lower = np.array([94, 80, 2], np.uint8)

blue\_upper = np.array([120, 255, 255], np.uint8)

blue\_mask = cv2.inRange(hsvFrame, blue\_lower, blue\_upper)

# Morphological Transform, Dilation

# for each color and bitwise\_and operator

# between imageFrame and mask determines

# to detect only that particular color

kernel = np.ones((5, 5), "uint8")

# For red color

red\_mask = cv2.dilate(red\_mask, kernel)

res\_red = cv2.bitwise\_and(imageFrame, imageFrame,mask = red\_mask)

# For green color

green\_mask = cv2.dilate(green\_mask, kernel)

res\_green = cv2.bitwise\_and(imageFrame, imageFrame,mask = green\_mask)

# For blue color

blue\_mask = cv2.dilate(blue\_mask, kernel)

res\_blue = cv2.bitwise\_and(imageFrame, imageFrame,mask = blue\_mask)

# Creating contour to track red color

contours, hierarchy = cv2.findContours(red\_mask,cv2.RETR\_TREE,cv2.CHAIN\_APPROX\_SIMPLE)

for pic , contour in enumerate(contours):

area = cv2.contourArea(contour)

if(area > 300):

x, y, w, h = cv2.boundingRect(contour)

imageFrame = cv2.rectangle(imageFrame, (x, y),(x + w, y + h),(0, 0, 255), 2)

cv2.putText(imageFrame, "Red Colour", (x, y),cv2.FONT\_HERSHEY\_SIMPLEX, 1.0,(0, 0, 255))

# Creating contour to track green color

contours, hierarchy = cv2.findContours(green\_mask,cv2.RETR\_TREE,cv2.CHAIN\_APPROX\_SIMPLE)

for pic, contour in enumerate(contours):

area = cv2.contourArea(contour)

if(area > 300):

x, y, w, h = cv2.boundingRect(contour)

imageFrame = cv2.rectangle(imageFrame, (x, y),(x + w, y + h),(0, 255, 0), 2)

cv2.putText(imageFrame, "Green Colour", (x, y),cv2.FONT\_HERSHEY\_SIMPLEX,1.0, (0, 255, 0))

# Creating contour to track blue color

contours, hierarchy = cv2.findContours(blue\_mask,cv2.RETR\_TREE,cv2.CHAIN\_APPROX\_SIMPLE)

for pic, contour in enumerate(contours):

area = cv2.contourArea(contour)

if(area > 300):

x, y, w, h = cv2.boundingRect(contour)

imageFrame = cv2.rectangle(imageFrame, (x, y),(x + w, y + h),(255, 0, 0), 2)

cv2.putText(imageFrame, "Blue Colour", (x, y),cv2.FONT\_HERSHEY\_SIMPLEX,1.0, (255, 0, 0))

# Program Termination

cv2.imshow("Multiple Color Detection in Real-TIme", imageFrame)

if cv2.waitKey(10) & 0xFF == ord('q'):

cv2.release()

cv2.destroyAllWindows()

break

def imgto():

img = cv2.imread(os.getcwd()+"\\"+"example.png")

#1

#height, width, \_ = img.shape

# square root of sums of squares of the sides is the actual max

# (Pythagoras Theorem) but that requires importing the math module

max\_distance = 1000

print("Press Esc to exit the program! :)")

# declaring global variables (are used later on)

clicked = False

r = g = b = xpos = ypos = 0

# Reading csv file with pandas and giving names to each column

index = ["color", "color\_name", "hex", "R", "G", "B"]

csv = pd.read\_csv("colors.csv", names=index, header=None)

# function to calculate minimum distance from all colors and get the most matching color

def getColorName(R, G, B):

minimum = max\_distance

for i in range(len(csv)):

d = (

abs(R - int(csv.loc[i, "R"]))

+ abs(G - int(csv.loc[i, "G"]))

+ abs(B - int(csv.loc[i, "B"]))

)

if d <= minimum:

minimum = d

cname = csv.loc[i, "color\_name"]

return cname

# function to get x,y coordinates of mouse double click

def draw\_function(event, x, y, flags, param):

if event == cv2.EVENT\_LBUTTONDOWN:

global b, g, r, xpos, ypos, clicked

clicked = True

xpos = x

ypos = y

b, g, r = img[y, x]

b = int(b)

g = int(g)

r = int(r)

# driver code

cv2.namedWindow("Photo")

cv2.setMouseCallback("Photo", draw\_function)

while True:

cv2.imshow("Photo", img)

if clicked:

cv2.rectangle(img, (20, 20), (750, 60), (b, g, r), -1)

colorName = getColorName(r, g, b) + " R=" + str(r) + " G=" + str(g) + " B=" + str(b)

cv2.putText(img, colorName, (50, 50), 2, 0.8, (255, 255, 255), 2, cv2.LINE\_AA)

if r + g + b >= 600:

cv2.putText(img, colorName, (50, 50), 2, 0.8, (0, 0, 0), 2, cv2.LINE\_AA)

clicked = False

# Exit Program if esc key is pressed

if cv2.waitKey(20) & 0xFF == 27:

break

cv2.destroyAllWindows()

print("Press 1 for Image to color detection \n 2.Press 2 for live cam detection")

n=int(input())

if(n==1):

imgto()

elif(n==2):

camto()

else:

print("wrong number")