A

Project Report

On

"Color Analyser"

Sagar Siddharth Waghmare

Roll Number: 4245

Rushikesh Daulat Pawar

Roll Number: 4266

B.E. (B)

Seminar Guide

Prof. M. B. Vaidya

For the partial fulfilment of
Mini Project Work in B.E. Computer Engineering

Department of Computer Engineering
Amrutvahini College of Engineering
Sangamner

2021-22

Department of Computer Engineering

Amrutvahini College of Engineering, Sangamner

CERTIFICATE

This is to certify that **Mr. Sagar Siddharth Waghmare and Mr. Rushikesh Daulat Pawar** student in Final Year Division B, Computer Engineering has successfully completed his Mini Project titled **"Color analyser"** at Amrutvahini College of Engineering, Sangamner towards partial fulfillment of Mini Project Work in Computer Engineering.

Prof. M. B. Vaidya
Project Guide

M. A. Wakchaure
Project Coordinator

Prof. R. L. Paikrao
Head of Department

Dr. M. A. Venkatesh
Principal

Abstract

The main objective of this application is the methodology for identifying the shades of colors with an exact prediction with their names. A study says, a normal human can able to clearly identify nearly 1 million shades of colors. But in the case of human having "enchroma", could be able to see only 1% (i.e.10,000 colors) from the normal humans. While painting pictures, a painter needs to identify the color patterns exactly or else the reality of image is not clear.
Keywords *-analyser, Shades, CV2*

Acknowledgements

Achievement is Finding out what you have been doing and what you have to do. The higher is submit,the harder is climb. The goal was fixed and I began with the determined resolved and put in a ceaseless sustained hard work. Greater the challenge, greater was our determination and it guided us to overcome all difficulties. It has been rightly said that we are built on the shoulders of others. For everything I have achieved, the credit goes to who had really help us to complete this seminar and for the timely guidance and infrastructure. Before we proceed any further, We would like to thank all those who have helped me in all the way through. To start with I thank my guide Prof. M. B. Vaidya, for there guidance, care and support, which she offered whenever I needed it the most. I would also like to take this opportunity to thank to project Coordinator M. A. Wakchaure and our respected Head of Department Prof. R. L. Paikrao. I also thankful to Honorable Principal Dr. M. A. Venkatesh sir for his encouragement and support.

Contents

List of Figures

Abbreviations

RGB : Red Green Blue

Chapter 1

Introduction

1.1 Introduction

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 perform 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 co-ordination 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 isuseful in detecting color names in this project.

Three different colors Red, Green and Blue are being tracked by utilizing the fundamentals of computer vision. After successful compilation when we execute the code a window redirectsto the image displayed on it whose path is given as an argument.

Additionally, we obtain the color name of the pixel along with the composition of three different colors red, blue and green values. It is helpful in recognizing colors and in robotics. One of the applications of color detection bycomputer vision is in driver less cars. This system is useful in detecting traffic and vehicle backlights and takes decision

to stop, start and continue driving. This also have much application in industry to pick and place different colored object by the robotic arm. Color detection is also used as a tool in various image editing and drawing apps.

Chapter 2

SYSTEM ARCHITECTURE

2.1 METHODS AND MATERIAL:

2.1.1 Take Image

The first step is to fetch a high-quality image with resolution and give it as a input to open CV2

2.1.2 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"]

2.1.3 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
ext = getColorName(r,g,b) + 'R=' + str(r) + 'G=' + str(g) + 'B=' + str(b).

Chapter 3

Output & Source Code

3.1 Source Code

```
#                    DA mini project
#  Project name -    Colour Analyser
#  Guided by    -    Prof. Milind Vaidya
#  Developed by -    (4266) Rushikesh Daulat pawar
#                    (4245) Sagar Siddharth Waghmare

import cv2
import pandas as pd

# Reading the image
img = cv2.imread(r'C:\Users\Rishi\PycharmProjects\colour Analyser\sample.
                                        jpg')

# Variable Declaration
clicked = False
r = g = b = xpos = ypos = 0

# Taking colors data as input using Pandas
index = ['colors', 'color-names', 'hex-value', 'R-value', 'G-value', 'B-
                                        value']
df = pd.read_csv('colors.csv', names=index, header=None)
```

```
# Function for selecting color of selected point by double clicking the
                                    left button of mouse
def selectColor(event, x, y, flag, param):
    if event == cv2.EVENT_LBUTTONDBLCLK:
        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)


# Taking a window to display an Image
cv2.namedWindow('Sample')

# Callback function for mouse events
cv2.setMouseCallback('Sample', selectColor)


# Function for getting color name of selected area
def getColorName(R, G, B):
    global colorName
    minimum = 10000

    #  calculate a distance(d) which tells us how close we are to color
                                    and choose the one having minimum
                                    distance.
    for i in range(len(df)):
        d = abs(R - int(df.loc[i, 'R-value'])) + abs(G - int(df.loc[i, 'G
                                    -value'])) + abs(B - int(df.loc[i, '
                                    B-value']))
        if d <= minimum:
            minimum = d
            colorName = df.loc[i, "color-names"]

    return colorName

# Updates the color name whenever the double click occurs
```

```
while 1:

    # We shown the image window
    cv2.imshow('Sample', img)
    if clicked:
        cv2.rectangle(img, (20, 20), (750, 60), (b, g, r), -1)
        text = getColorName(r, g, b)
        cv2.putText(img, text, (50, 50), 2, 0.8, (255, 255, 255), 2, cv2.
                                    LINE_AA)

        if r + g + b >= 500:
            cv2.putText(img, text, (50, 50), 2, 0.8, (0, 0, 0), 2, cv2.
                                    LINE_AA)

        clicked = False

    # Exits when the user presses the 'Esc' button
    if cv2.waitKey(20) & 0xFF == 27:
        break

# Clears all the windows
cv2.destroyAllWindows()
```

3.2 Output snapshots

Snap 1

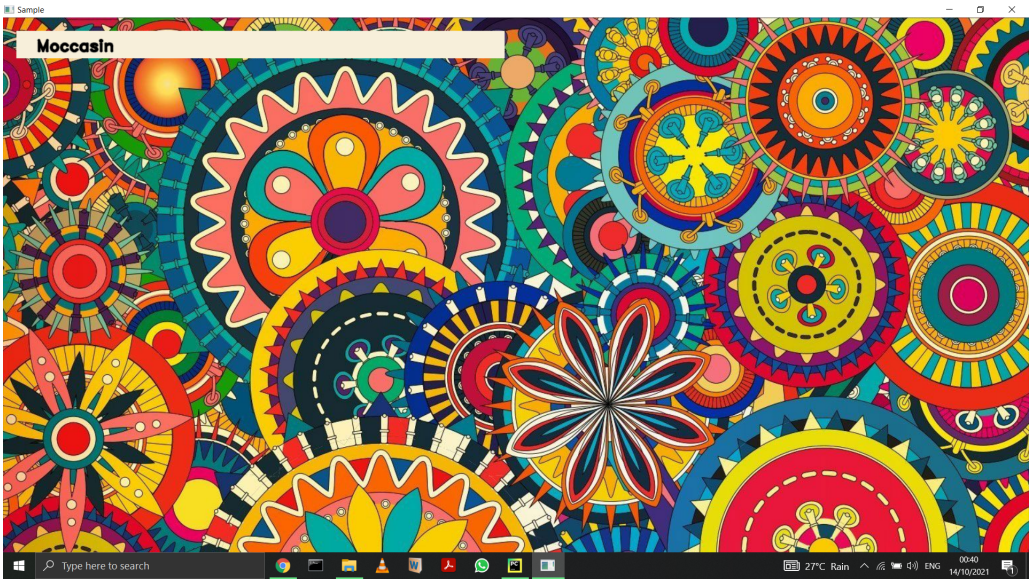

FIGURE 3.1: Snap 1 - Color (Moccasin)

Snap 2

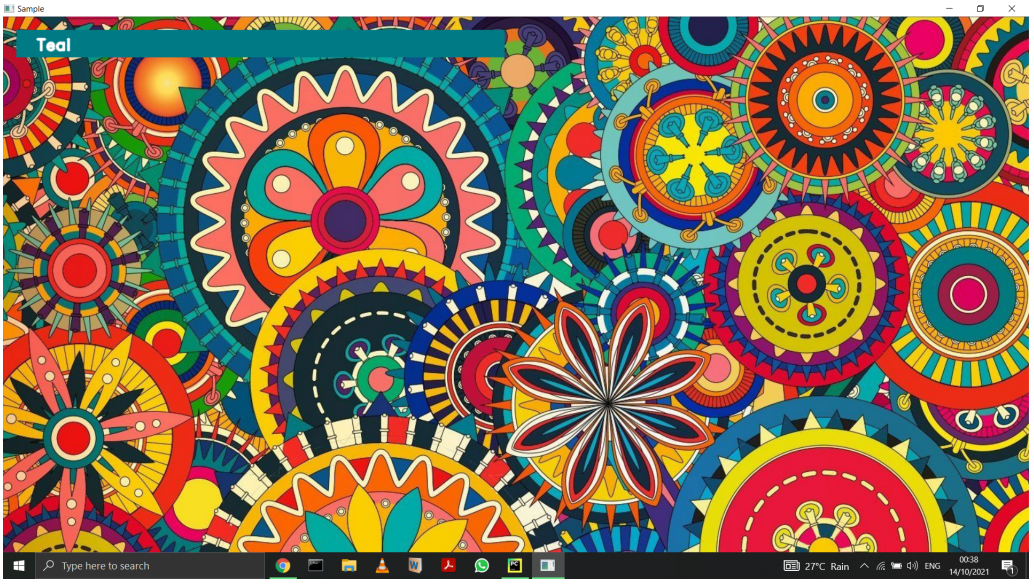

FIGURE 3.2: Snap 2 - Color (Teal)

Snap 3

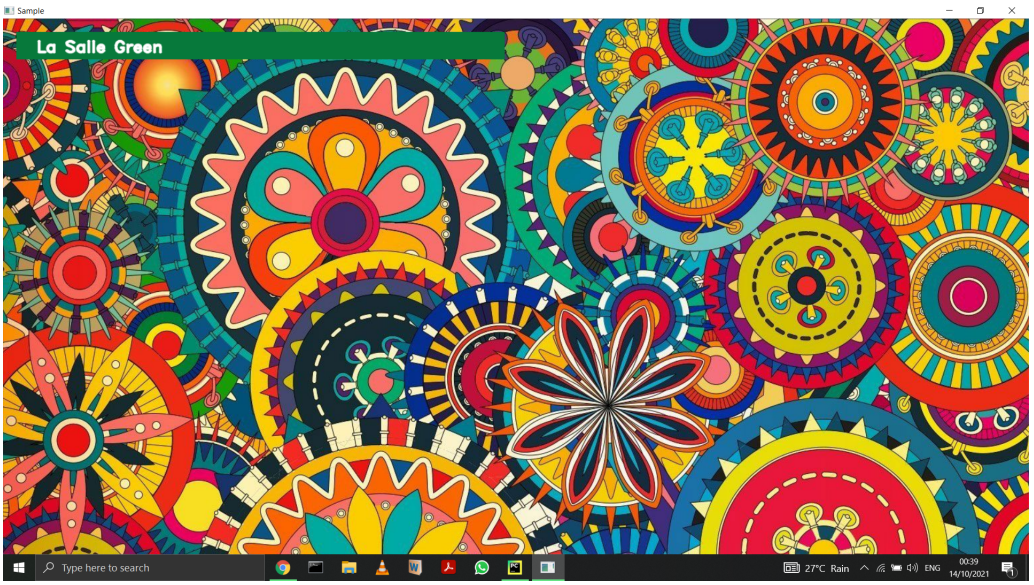

FIGURE 3.3: Snap 3 - Color (La Salle Green)

Snap 4

FIGURE 3.4: Snap 4 - Color (Deep Carmine Pink)

Snap 5

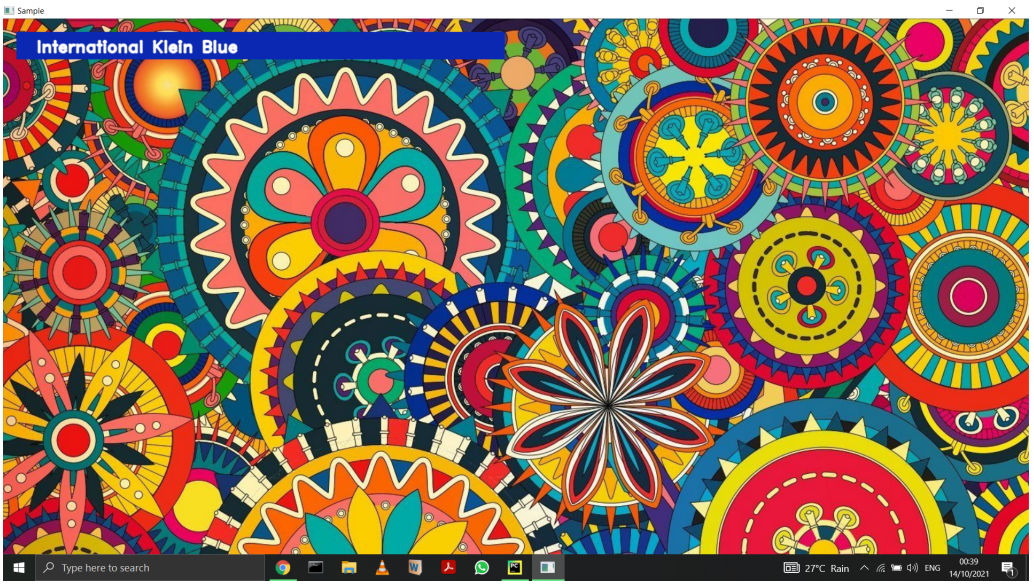

FIGURE 3.5: Snap 5 - Color (International Klein blue)

Chapter 4

CONCLUSION

In this system 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.

References

[1] 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

[2] G.M. Snoek, "Evaluating Color Descriptors for Object and Scene Recognition", IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 32, NO. 9, SEPTEMBER 201