Bahria University,

Karachi Campus

Course: CSL-411 – Artificial Intelligence Lab
Term: Fall 2020, Class: BSE- 5(A & B)

PROJECT REPORT

Submitted By:

- **Gul Saba (02-131182-009) – BSE-5B**
- **Sundas Hafeez (02-131182-025)**
- **Sadia Afzal (02-131182-022)**
- **Anusha Saeed (02-131182-076)**

Submitted To:

Engr . Muhammad Rehan Baig

Submitted On:

23-01-2020

INDEX

Title :

COLOUR DETECTION

- **ABSTRACT:**

The goal we set for our AI project experiment was to detect the color of the dominant object of a picture. Searching by color is a common occurrence when it comes to e-commerce websites, and while the nature of the object is usually clear in the description, its color is often more of a problem. As an example, let us consider an article containing several colors in its description (for the sake of the example: white and blue), but having a picture clearly showcasing a blue shirt. While searching for a white shirt may show up and may even rank very high, because the relevant color. However, correct this may be, having a number of images showing a blue object while searching for a white one does not give the best impression of relevance to the end user. The idea is certainly not to remove these results (because they are still correct), but to boost the ranking of the ones which more closely represent objects of the searched color.

- **INTRODUCTION:**

We will be working with colors and learn about many concepts throughout this project. Color detection is necessary to recognize objects, it is also used as a tool in various image editing and drawing apps. In this color detection Python project, we build an application through which you can automatically get the name of the color by clicking on them. So, for this, we have a data file that contains the color name and its values. We calculate the distance from each color and find the shortest one.

- **OBJECTIVE:**

The main objective of the project is to build a color detector that can classify colors and its values. To build a color detector that can classify colors machine learning is needed. The goal is to create a model and train the model with large datasets and test its accuracy in detecting color. If the model can classify color with good accuracy rate then we have successfully built a color detector. The color detector must have good accuracy in detecting color. The color detector must be trained with large datasets. This color detector will be used for building real applications.

• DESCRIPTION:

The programming language which is used to build this color detector project is python and the Pandas and OpenCV python AI libraries are used.

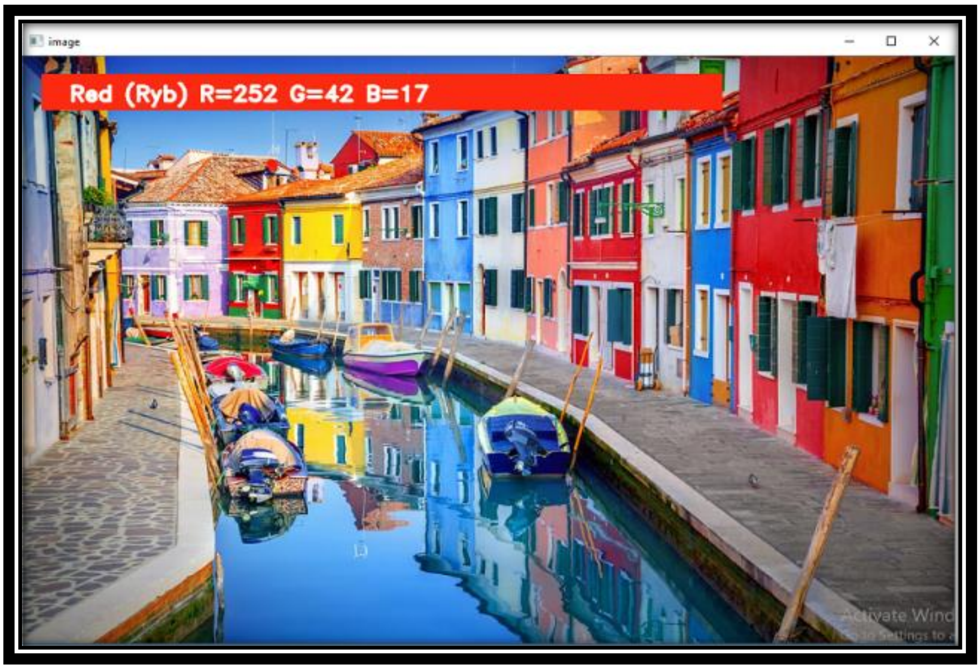
DATASET:

Colors are made up of 3 primary colors; red, green, and blue. In computers, we define each color value within a range of 0 to 255. So, in how many ways we can define a color? The answer is 256*256*256 = 16,581,375. There are approximately 16.5 million different ways to represent a color. In our dataset, we need to map each color’s values with their corresponding names. But we used a dataset that contains RGB values with their corresponding names.

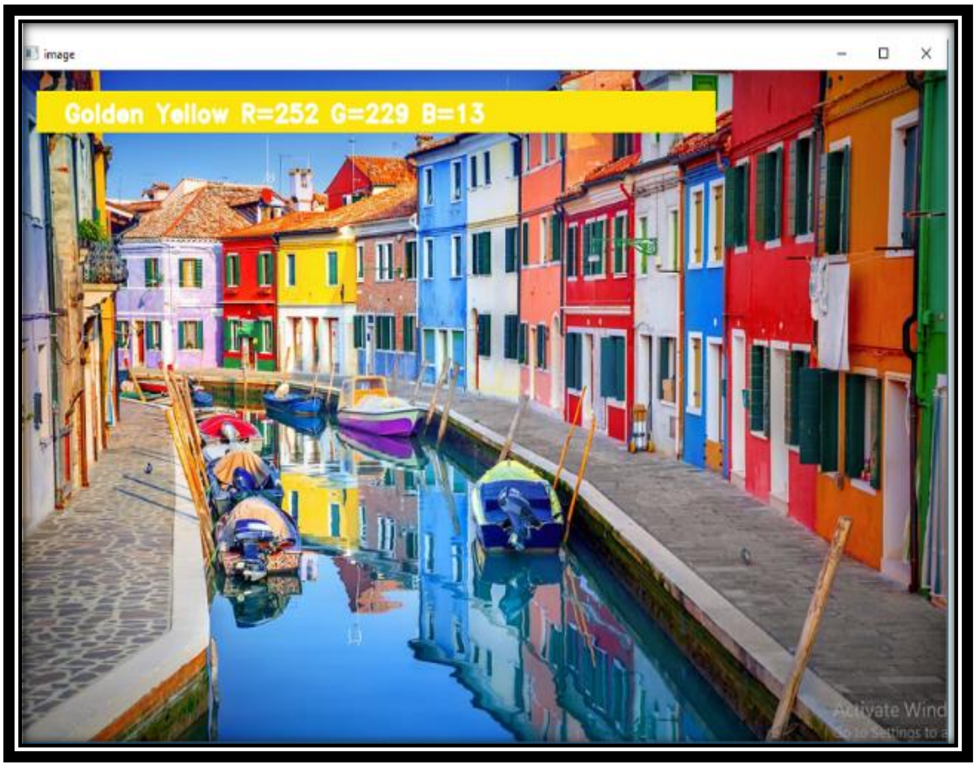
• CODE:

```
import cv2
import numpy as np
import pandas as pd
import argparse

ap = argparse.ArgumentParser()
ap.add_argument('-i', '--image', required=True, help="Image Path")
args = vars(ap.parse_args())
img_path = args['image']

img = cv2.imread(img_path)

clicked = False
r = g = b = xpos = ypos = 0

index=["color","color_name","hex","R","G","B"]
csv = pd.read_csv('colors.csv', names=index, header=None)

def getColorName(R,G,B):
    minimum = 10000
    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

def draw_function(event, x,y,flags,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)

cv2.namedWindow('image')
cv2.setMouseCallback('image',draw_function)

while(1):

    cv2.imshow("image",img)
    if (clicked):


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


        text = getColorName(r,g,b) + ' R='+ str(r) +  ' G='+ str(g)
+  ' B='+ str(b)


        cv2.putText(img,
text,(50,50),2,0.8,(255,255,255),2,cv2.LINE_AA)


        if(r+g+b>=600):
            cv2.putText(img,
text,(50,50),2,0.8,(0,0,0),2,cv2.LINE_AA)

        clicked=False


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

cv2.destroyAllWindows()
```

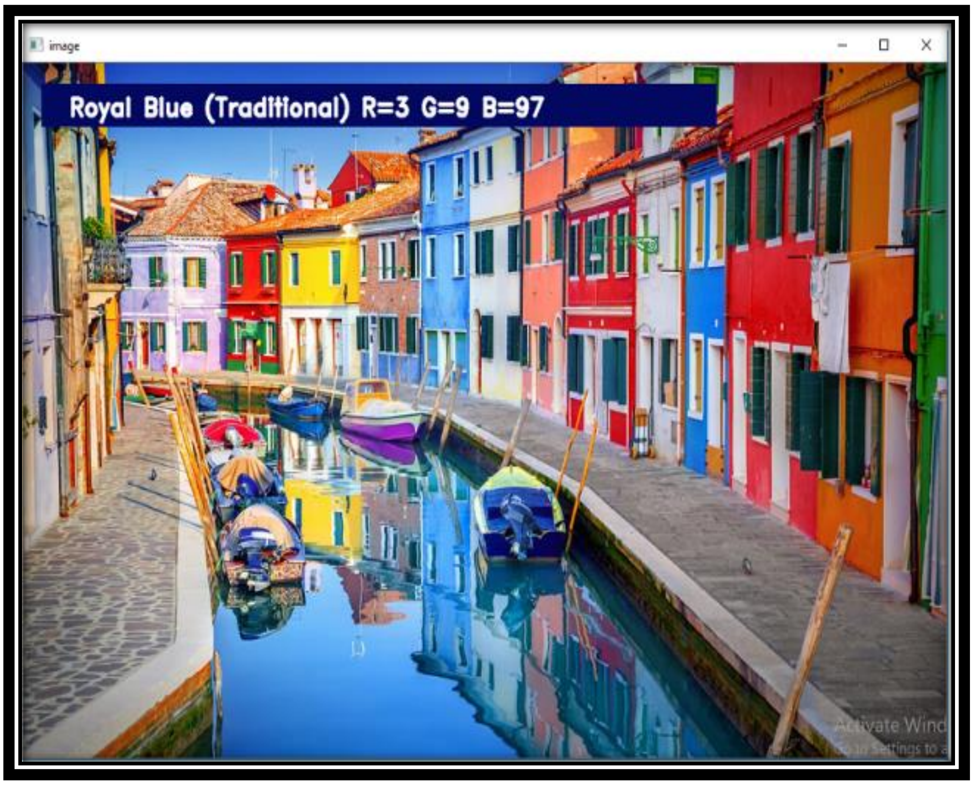
- **OUTPUT:**



```
Red (Ryb) R=252 G=42 B=17
```



```
Golden Yellow R=252 G=229 B=13
```



Royal Blue (Traditional) R=3 G=9 B=97

- **<u>CONCLUSION:</u>**

In this Python project, we learned about colors and how we can extract color RGB values and the color name of a pixel. We learned how to handle events like double-clicking on the window and saw how to read CSV files with pandas and perform operations on data. This is used in numerous images editing and drawing apps.

- **<u>GIT LINK:</u>**

You can feel free to access our project from this link:

<u>https://github.com/sundashafeez1/colordetection</u>

---------- x ---------- x ----------