Color Detection using Python and OpenCV

An Application Development – 1 (Project) Report Submitted
In partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering (Data Science)

by

CH.NagaSri	**20N31A6711**
A.Dheeraj	**20N31A6703**
MD.Khaleel	**21N35A6701**

Under the Guidance of

Mr.M.Venu
Associate Professor
Department of Emerging Technologies
MRCET (Autonomous Institution, UGC Govt. of India)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(EMERGING TECHNOLOGIES)
MALLA REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY
(Autonomous Institution – UGC, Govt. of India)
(Affiliated to JNTU, Hyderabad, Approved by AICTE, Accredited by NBA & NAAC – 'A' Grade, ISO 9001:2015 Certified)
Maisammaguda (v), Near Dullapally, Via: Kompally, Hyderabad – 500 100, Telangana State, India

2022-2023

MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY
(Autonomous Institution – UGC, Govt. of India)
(Sponsored by CMR Educational Society)
Recognized under 2(f) and 12 (B) of UGC ACT 1956
Estd : 2004
(Affiliated to JNTUH, Hyderabad, Approved by AICTE- Accredited by NBA & NAAC– 'A' Grade - ISO 9001:2015 Certified)

CERTIFICATE

This is to certify that this is the bonafide record of the project titled **"Color Detection using Python and OpenCV"** submitted by **CH.Nagasri(20N31A6711) A.Dheeraj(20N31A6703) and MD.Khaleel(21N35A6701)** of **B.Tech III Year – I Semester** in the partial fulfillment of the requirements for the degree of **Bachelor of Technology** in **Computer Science and Engineering (Specialization)**, Dept. of CSE (Emerging Technologies) during the year 2022-2023. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide
Mr.M.Venu
Associate professor
Department of CSE(ET)

Project Coordinator
Dr. P Dileep
Professor
Department of CSE(ET)

EXTERNAL EXAMINER

Dr. M V Kamal &
Head of the Department

DECLARATION

We hereby declare that the project entitled **lor Detection using Python and OpenCV"** submitted to **Malla Reddy College of Engineering and Technology,** affiliated t**o** Jawaharlal Nehru Technological University Hyderabad (JNTUH) as part of III Year B.Tech – I Semester and for the partial fulfillment of the requirement for the award of **Bachelor of Technology** in **Computer Science and Engineering (DataScience)** is a result of original research work done by us.

It is further declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of degree or diploma.

CH. NAGASRI	**-**	**20N31A6711**
A. DHEERAJ	**-**	**20N31A6703**
MD.KHALEEL	**-**	**21N35A6701**

ACKNOWLEDGEMENT

We feel myself honored and privileged to place our warm salutation to my college "Malla Reddy College of Engineering and Technology (Autonomous Institution – UGC Govt. of India) and our Principal **Dr. S Srinivasa Rao,** Professor who gave us the opportunity to do the Application Development -1 (Project) during my III Year B.Tech and profound the technical skills.

We express my heartiest thanks to our Director **Dr. V S K Reddy,** Professor for encouraging us in every aspect of my project and helping me realize my full potential.

We also thankful to our Head of the Department **Dr. M V Kamal,** Professor for providing training and guidance, excellent infrastructure and a nice atmosphere for completing this project successfully.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr.M.Venu,** for his valuable suggestions and interest throughout the course of this project.

We convey our heartfelt thanks to our Project Coordinator **Dr. P Dileep,** Professor for allowing for their regular guidance and constant encouragement during our dissertation work.

We would like to thank all my supporting **staff** of the Department of CSE (Emerging Technologies) and even all other department who have been helpful directly and in-directly in making our project a success.

Finally, we would like to take this opportunity to thank our **families** for their support and blessings for completion of our project that gave me the strength to do our project.

CH.Nagasri(20N31A6711)
A.Dheeraj (20N31A6703)
MD.Khaleel(21N35A6701)

ABSTRACT

The main objective of color detection is the methodology for identifying the shades of colors with an exact prediction with their names. In this method the color codes already provided in the program are compared it with the image whose colors we want to know. A study says that 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. In this project we are using opencv, pandas with python.

TABLE OF CONTENTS

Chapter No.		Contents	Page no
1		Introduction	1
	1.1	Problem Definition	2
	1.2	Existing System	3
	1.3	Proposed System	4
2		System Requirements	5
	2.1	Software Requirements	6
	2.2	Hardware Requirements	8
3		System Design	9
	3.1	Dataflow Diagrams / UML Diagrams	10
	3.2	Modules	12
4		Implementation	13
5		Results and Sample Output	19
6		Conclusion & Future Scope	22
7		References	24

LIST OF FIGURES

S.No	Figure Title	Page No
1.	System Architecture	9
2.	Data Flow Chart	10
3.	User Case Diagram	11
4.	Output Screen	19-21

CHAPTER 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 is useful 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 redirects to 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 by computer 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.

1.1 PROBLEM DEFINITION

Over the time, academic computer engineers, theoretical vision researchers and commercial product developers have enhanced the performance of automatic color detection algorithms in the spread of speedy processors and algorithms to detect the colors. This is what is required and have been implemented Since people are currently interested in quick and rapid operations in many real-world situations. An important challenge in color detection and similar technologies is the ability to handle all those situations where subjects (whose color is to be detected) are not in close range and hence color cannot be differentiated and hence detection becomes tough and may lead to wrong results or no results. There are many factors that make the color detection easy or tough.

These sources of variance are divided into two groups: Internal and External Features

Internal factors: - are caused by the sensors that is the camera of the system and are independent of the observer

External factors: - make effect on image coming to the camera with the combination or interaction of the sunlight, rain, dirt storm etc or even due to dirt on camera

1.2: EXISTING SYSTEM

Existing system is based on the RGB colour model. RGB color model represents colour in red(R), green(G) and blue(B) components. In the current system, we can only detect the 3 colours in the image that is related to blue or green or red. In existing system there is no exact color representation of colors with accuracy. It cannot detect all the colors in the image.

1.3: PROPOSED SYSTEM

In proposed system, we are introducing the CV datasets and according to it the number of shades that can be identified using color names along with their RGB values. Whenever the cursor clicks the image, it automatically shows the color name and RGB color values. Proposed system uses OpenCv for sorting of primary colors. Colour detection is made to reach each and everyone in the society suffering from color blindness so that everyone can get benefit from this.

CHAPTER 2:
SYSTEM REQUIREMENTS

Reqirement analysis is involved in defining customer needs and objectives in the context of planned customer use, and system features identified to determine the needs of system functions.

User Requirements: -

User engagement with factual statements and assumptions that define system expectations based on objectives, environment, issues and measures of efficiency and effectiveness.

Basically, users need:

i) A system that improves the efficiency of data retention and retrieval.

ii) An easy-to-read and use system.

iii) A system that speeds up the processing of transactions.

iv) A flexible, secure and flexible system.

2.1: SOFTWARE REQUIREMENTS

Languages used: -

- Python
- Python is a high-level, general purpose computer language. Python can be used on a server to create web applications. Python works on different platforms (Windows, Mac, Linux, Raspberry pi,etc).Python runs on a interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Module Used: -

- **Pandas**
- Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data.
- **OpenCV**
- OpenCV (*Open Source Computer Vision Library*) is a huge open-source library for computer vision, machine learning and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. When it is integrated with various libraries,such as Numpy which is a highly optimized library for numerical operations, then the number of weapons increases in your arsenal i.e whatever operations one can do in numpy can be combined with Opencv

Role of OpenCv:

The Role of OpenCV is an open source 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. OpenCV is highly rated as it involves state of Art Computer Vision and Machine Learning algorithms, Deep Learning etc. Deep Learning helps in self driving cars and those cars will be using OpenCV to gather colors either be it on road traffic lights or other cars light, it may can passing light, stop light or indicator lights.

2.2 HARDWARE REQUIRMENTS

1. Ram: 4GB or more.
2. Processor: i3 7th gen or more.
3. Programming Language: Python 3.6.
4. Hard disk space: 10GB or more.
5. Mouse

CHAPTER 3

SYSTEM DESIGN

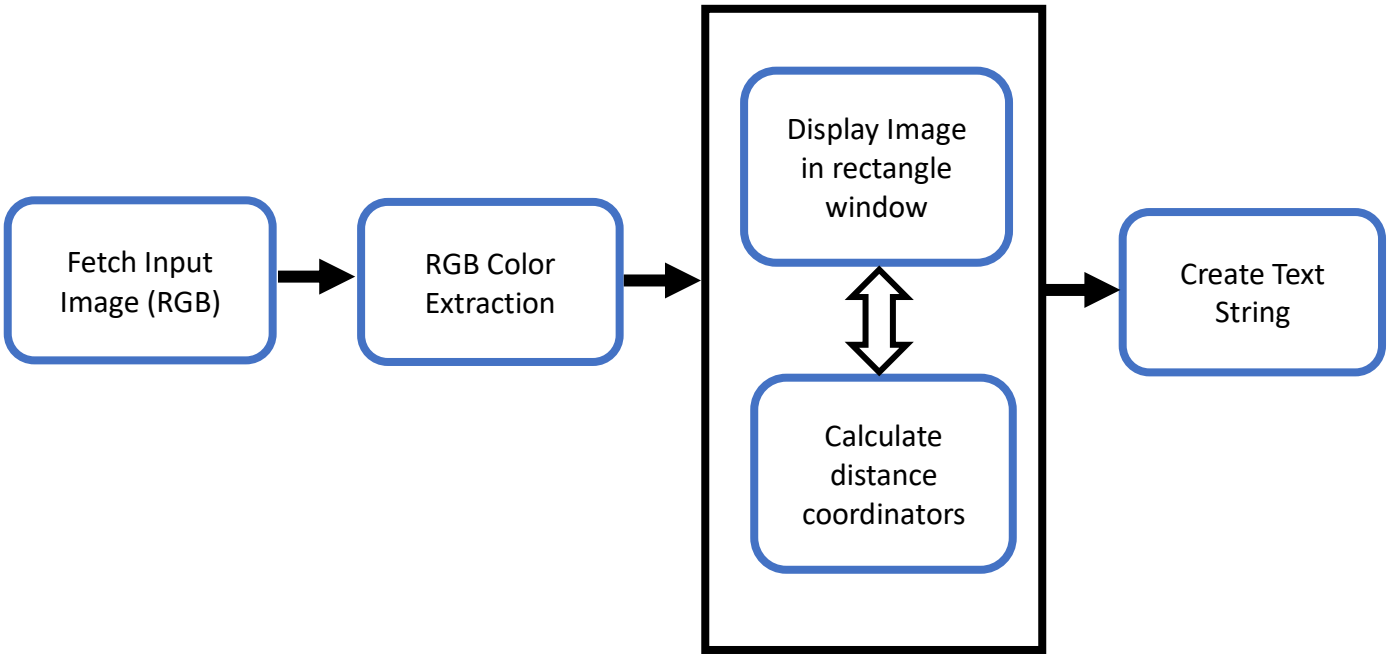


Fig 1. Architecture

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 python. The above architecture makes the process more efficient based on principles and properties related to each other. As we know that 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 colors red, green and blue in the given image.

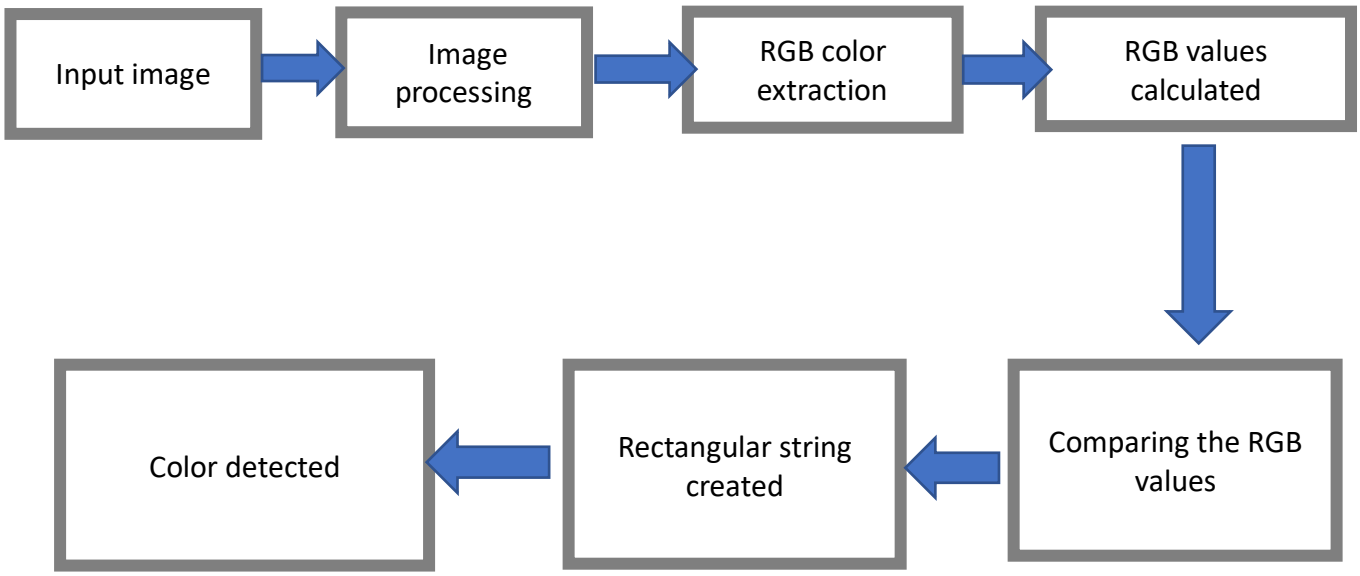


Fig 2.Data Flow Diagram

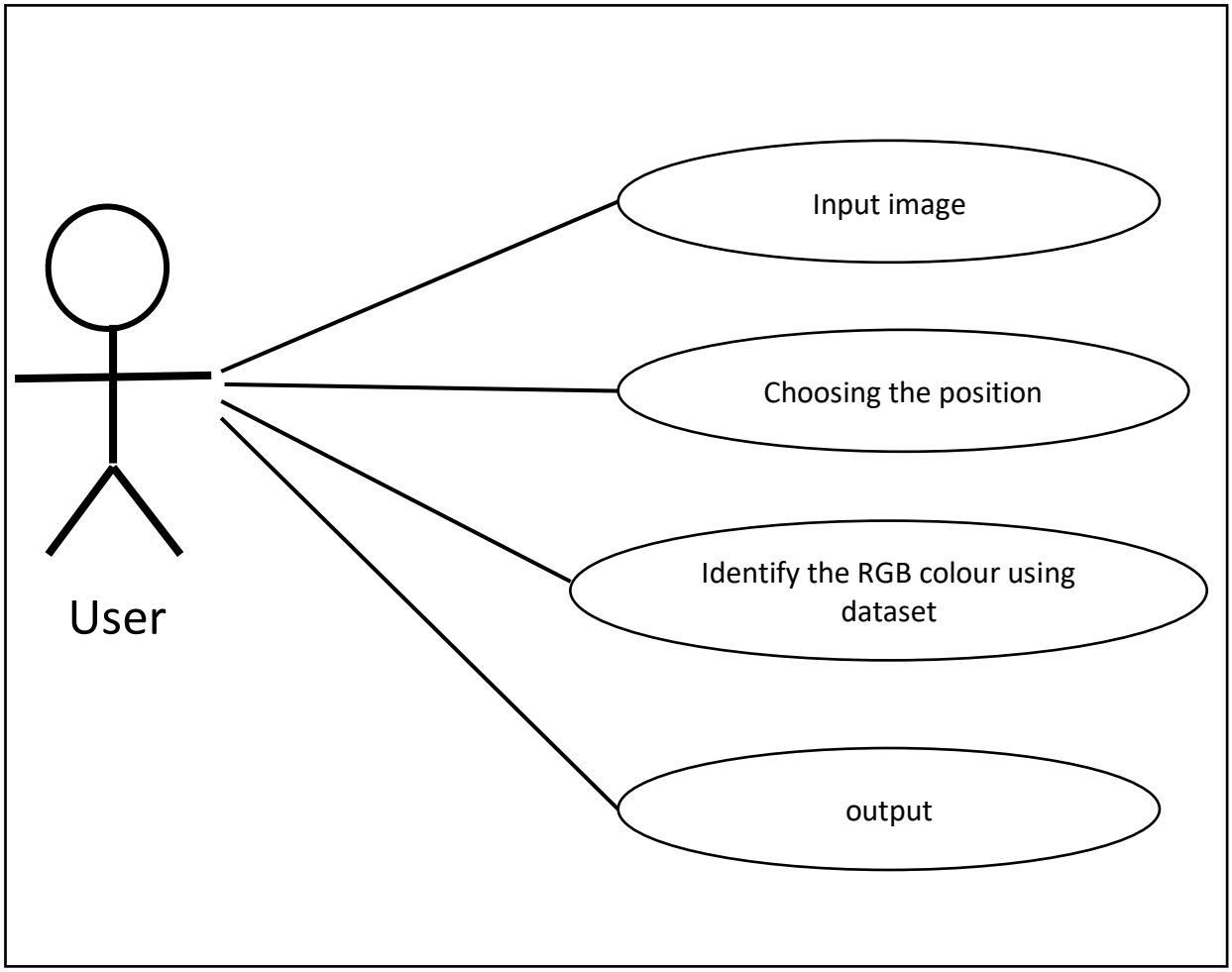


Fig 3. User Case Diagram

3.2 MODULES

Module 1: Capturing and Storing Image :

In this module, the capturing of image takes place. The image stored in this process is later used for detecting the color. The program stores the image and resizes it to 800*600 pixels of image the user gives in as input. The reduction of image size and leads to judicial use of the storage provided

Module II: Image Processing :

In this module, when the program is executed the program opens a window of size 800*600 image pixel are processed an the program loads the image in the window. In this module, the programs run over gives the image for further operation. When all the images are perfectly processed assnd the program is ready to detect the color.

Module III: Color Detection :

In this module, the programs has completed the Image Processing and is ready to take the input of the user. The program which is already displaying the window can now be clicked anywhere and it will display the color present there. The user Clicks on the image anywhere of whose color he wants to know.

4.IMPLEMENTATION

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. In our dataset, we need to map each color's values with their corresponding names. In our dataset, we need to map each color's values with their corresponding names. But don't worry, we don't need to map all the values. We will be using a dataset that contains RGB values with their corresponding names. The CSV file for our dataset has been taken from the colors.csv file include 865 color names along with their RGB and hex values.

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']
#python color_detection.py -i <add your image path here>
#Reading image with opencv
img = cv2.imread('colourpicture.jpeg')
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)
cv2.namedWindow('image')
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)
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
cv2.setMouseCallback('image',draw_function)
while(1):
    cv2.imshow("image",img)
    if (clicked):
        #cv2.rectangle(image, startpoint, endpoint, color, thickness) -1 thickness
fills rectangle entirely
        cv2.rectangle(img,(20,20), (750,60), (b,g,r), -1)
        #Creating text string to display ( Color name and RGB values )
        text = getColorName(r,g,b) + ' R='+ str(r) + ' G='+ str(g) + ' B='+ str(b)
        #cv2.putText(img,text,start,font(0-7), fontScale, color, thickness, lineType,
(optional bottomLeft bool) )
        cv2.putText(img, text,(50,50),2,0.8,(255,255,255),2,cv2.LINE_AA)
  #For very light colours we will display text in black colour
        if(r+g+b>=600):
            cv2.putText(img, text,(50,50),2,0.8,(0,0,0),2,cv2.LINE_AA)
        clicked=False
    #Break the loop when user hits 'esc' key
    if cv2.waitKey(20) & 0xFF ==27:
        break
cv2.destroyAllWindows()
```

Image Capture: The first step is to fetch a high-quality image with resolution. To load an image from a file we use Cv2.imread(). Image should be in working directory or full path of the image should be given.

Img=cv2.imread(img path)

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"]

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).

- **Step 1: Taking an image from the user.**
  Next, We are using ***argparse*** library to create an argument parser. We can directly give an image path from the command prompt

- **Step 2: Next, we read the CSV file with pandas.**
  Next, The pandas library is very useful when we need to perform various operations on data files like
  CSV. **pd.read_csv()** reads the CSV file and loads it into the pandas **DataFrame**. We have assigned each column with a name for easy accessing.

- **Step 3: Set a mouse callback event on a window.**
  Next, we created a window in which the input image will display. Then, we set a callback function which will be called when a mouse event happens.

- **Step 4: Create the draw_function.**
  It will calculate the rgb values of the pixel which we double click. The function parameters have the event name, (x,y) coordinates of the mouse position, etc.

  In the function, we check if the event is double-clicked then we calculate and set the r,g,b values along with x,y positions of the mouse.

- **Step 5: Calculate distance to get color name.**
  We have the r,g and b values. Now, we need another function which will return us the color name from RGB values.

To get the color name, we calculate a distance(d) which tells us how close we are to color and choose the one having minimum distance.

Our distance is calculated by this formula:
d = abs(Red – ithRedColor) + (Green – ithGreenColor) + (Blue – ithBlueColor)

- **Step 6: Display image on the window.**
  Whenever a double click event occurs, it will update the color name and RGB values on the window.
  Using the **cv2.imshow()** function, we draw the image on the window.

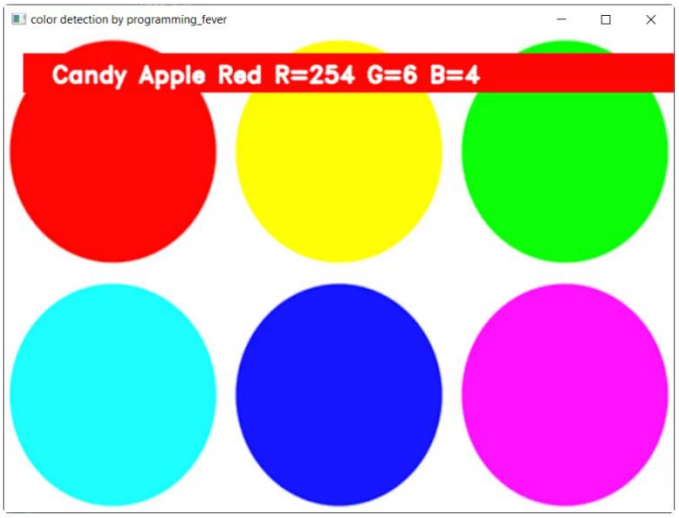
  When the user double clicks the window, we draw a rectangle and get the color name to draw text on thewindow
  using **cv2.rectangle** and **cv2.putText()** functions.

- **Step 7: Run Python File.**
  The beginner Python project is now complete, you can run the Python file from the command prompt. Make sure to give an image path using '**-i**' argument.If the image is in another directory, then you need to give full path of image.

5.RESULTS:

Fig(a)

Fig(b)

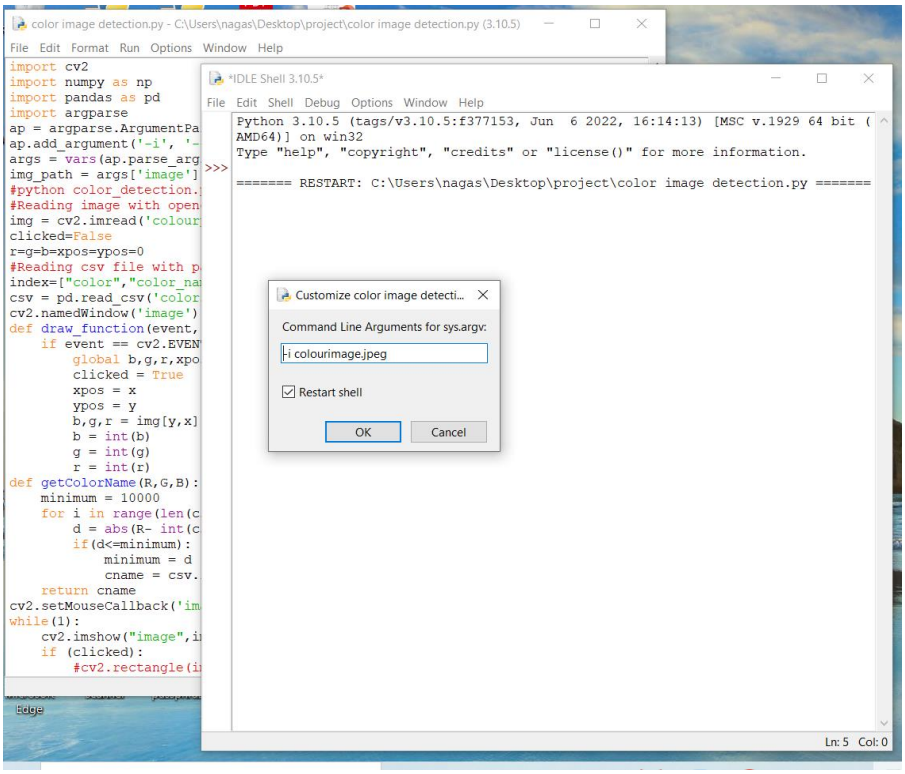

color detection by programming_fever
Candy Apple Red R=254 G=6 B=4

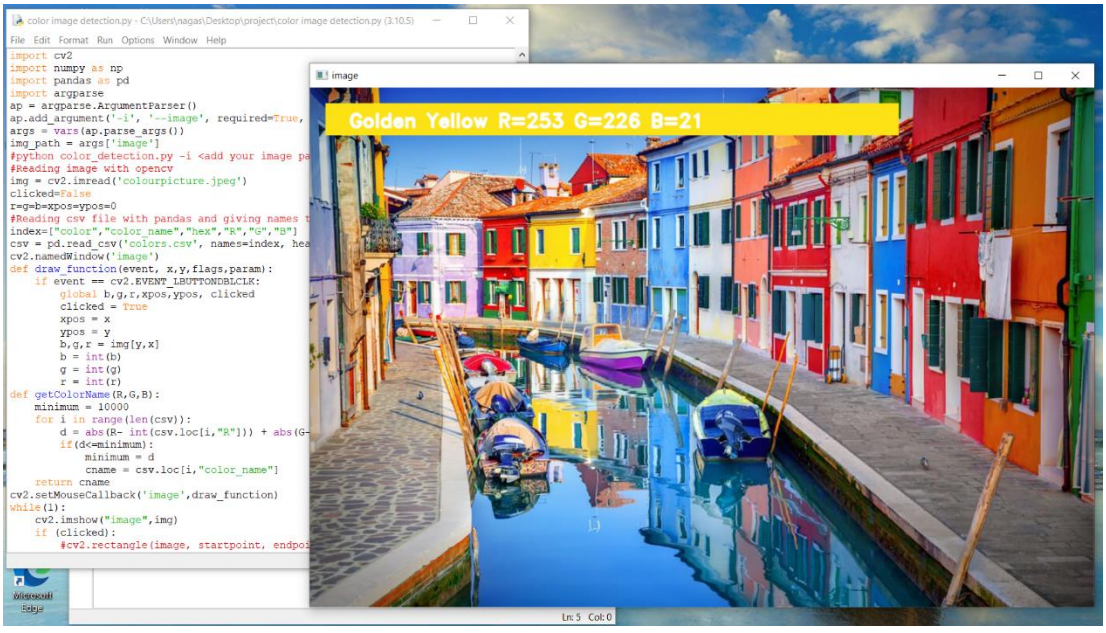
Fig(c)


Sapphire R=18 G=88 B=186

Fig(d)

SAMPLE OUTPUTS





6.CONCLUSION

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.

FUTURESCOPE

In existing system there is no exact color representation of colors with accuracy. In proposed system, we are introducing the CV datasets 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. Proposed system uses OpenCv for sorting of primary colors. The system can be modified to detect the colour of an image by capturing the live video. The system can be interfaced with a camera to detect the a wide range of colour shades

7.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 2010

[3]. 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

[4]. 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

[5]. 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

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