A Project Report

on

Color Observation using OpenCV

Submitted in partial fulfillment of the
requirement for the award of the degree of

BACHELOR'S OF TECHNOLOGY

Under The Supervision of
DR. E RAJESH
ASSOCIATE PROFESSOR

Submitted By

Sanidhya Sharan
19SCSE1180017
Vivek Abhijeet
19SCSE1180095

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
DECEMBER,2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"COLOR OBSERVATION USING OPENCV"** in partial fulfillment of the requirements for the award of the <u>Bachelors of Technology</u> submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of September 2021 to December 2021, under the supervision of Dr. E. Rajesh sir (Professor), Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Sanidhya Sharan,19SCSE1180017

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr. E Rajesh
Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Sanidhya Sharan,19SCSE1180017 has been held on _____ and his/her work is recommended for the award of Bachelors of Technology.

Signature of Examiner(s)   **Signature of Supervisor(s)**

Signature of Project Coordinator   **Signature of Dean**

Date: December, 2021 Place: Greater Noida

Acknowledgement

I cant communicate enough gratitude to my collegues for his proceeded with help and

consolation. Dr. E Rajesh sir, my venture direct who encouraged me to such an extent. I offer my

genuine thankfulness for the learning openings gave by resources.

Contents

List of figures

1	Workflow of process	
2	workflow	
3	Pictorial representation of open cv & python	
4	Experimental result	
5	Experimental result	
6	Experimental result	

Acronyms

B.Tech.	Bachelor of Technology
M.Tech.	Master of Technology
BCA	Bachelor of Computer Applications
MCA	Master of Computer Applications
B.Sc. (CS)	Bachelor of Science in Computer Science
M.Sc. (CS)	Master of Science in Computer Science
SCSE	School of Computing Science and Engineering

ABSTRACT:-

Detecting the name of a color is known as color detection. For Humans it is an easy thing to detect colors of different patterns but the same does not go with computers as computer being a machine cannot detect different colors easily. So, to overcome this problem we have chosen this project.

So, the main purpose or we can say objective of the methodology that we are using is to detect different color shades with exactly predicting their names. It has been studied that normally a human eye can identify somewhere around 1 million different color shades but, in few cases where people are having problems like "enchroma" They can only see 1% ie almost equal to 10,000 compared to a normal human eye.

When a painter paints a picture he needs to identify the patterns of color exactly or else the image is not presented clearly. So the topic that we have chosen many people have written research paper on this topic but we are using different procedure that is using Open CV for image processing and Pandas and Python language.

CHAPTER :-1

INTRODUCTION

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.Computer vision works much the same as human vision, except humans have a head start. Human sight has the advantage of lifetimes of context to train how to tell objects apart, how far away they are, whether they are moving and whether there is something wrong in an image.Computer vision trains machines to perform these functions, but it has to do it in much less time with cameras, data and algorithms rather than retinas, optic nerves and a visual cortex. Because a system trained to inspect products or watch a production asset can analyze thousands of products or processes a minute, noticing imperceptible defects or issues, it can quickly surpass human capabilities.So our main keyword here is OpenCV which is Open Computer Vision Library. So talking about its history it was first launched by intel in the year 1999 after that it has undergone lots of modification and updates to make it perfect for real time computer vision. C and C++ is the language which is used to write this library and we can operate this OPEN CV easily on our windows OS or mac or LINUX. Not only C and C++ but it can also

interface or can ve used with programming languages like MATLAB, Python, Ruby and many more. With the help of libraries like Numpy and language Python we can easily perform image processing (color detection).So talking about our project we are going to make an application where just by a click we can get the name of all the various colors present in our image. To successfully execute this we need a dataset that contains the name of all the colors present and their RGB values. Then when we click on the image part then it will calculate the distance from each color given in the dataset and find the shortest distance from one of the color. Here the basic colors that are RED GREEN & BLUE are being tracked by using the fundamental principles of CV(Computer Vision). After the coding is done and the we run the code i.e execute it then the output window is displayed or we can say that we are redirected to the output window and the respective image is showed whose path we have mentioned in our code while reading the image. After this we get the name of color of the pixel with the composition of the basic 3 colors values that are RGB (RED, BLUE AND GREEN). So one of its application can be its use in robotics where it can be used in detecting colors in driver less cars, there color detecting using our computer vision plays an important role as this thing help the cat detecting the traffic lights or backlights of vehicles and stop and move accordingly. Color detection can also be used in apps where we edit image and also in various drawing apps.

Color Model

The color model aims to facilitate the specifications of colors in some standard way.

Different types of color models are used in multiple fields like in hardware, in multiple applications of creating animation, etc.

- In Digital Image Processing, the hardware-oriented models that are commonly used are the RGB model for printers and color monitors.
- CMY(cyan, magenta, yellow) and CMYK(cyan, magenta, yellow, black) models are used for color printing.
- HSI(hue, saturation, intensity) deals with colors as humans interpret.

The 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 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 this link:
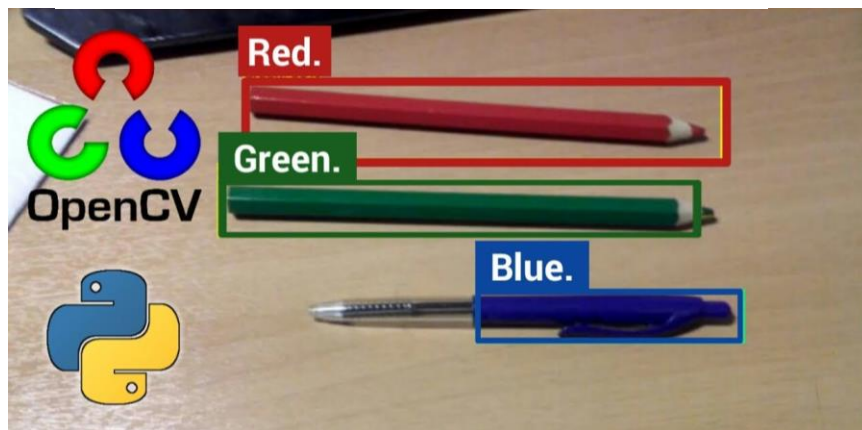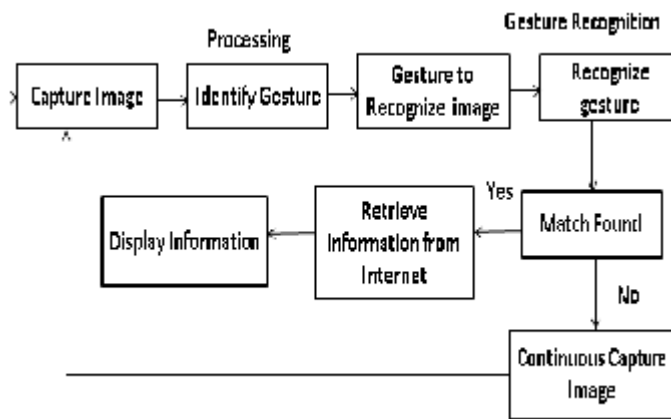
Colors Dataset

The colors.csv file includes 865 color names along with their RGB and hex values.

Prerequisites

Before starting with this Python project with source code, you should be familiar with the computer vision library of Python that is ***OpenCV*** and ***Pandas***.
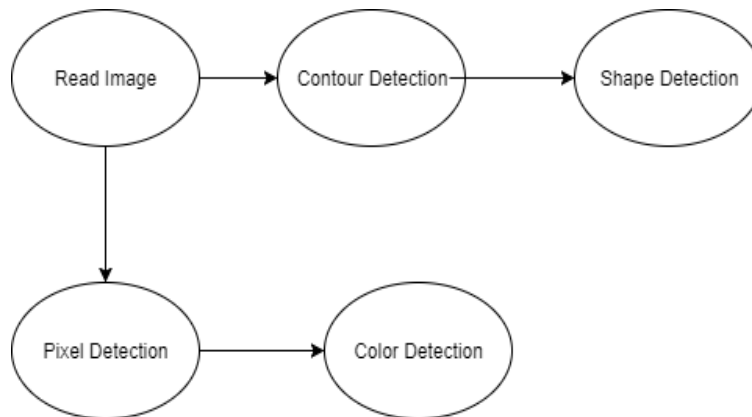
Representation of how it will be carried on

Chapter-2

Literature Survey/Project Design

P. Raguraman- 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.

Raghav Puri-Abstract - In the following work, we will be detecting contours, shapes and colors of various geometrical figures in the sample given binary images using Python 2.7, Open Source Computer Vision Library (OpenCV) and Numpy. The very fundamental functions are used for processing the images, that involves loading them, and detecting various shapes and colors inside the given sample images.

Hassan Jillani Farooqi-Color detection [1] is the process of detecting name of the color. Here this is easy task for humans to detect the color and choose one. But computer cannot detect the color easily. This is tough task for computer to detect the color easily. So that's why we

choose this project. Many of the project and research papers are written on this problem. But we use different techniques for this project

Yashasvee Thakur-—Color detection [1] is the process of detectingname of the color. Here this is easy task for humans to detect thecolor and choose one. But computer cannot detect the color easily. This is tough task for computer to detect the color easily.So that's why we choose this project. Many of the project and research papers are written on this problem. But we use different techniques for this project. Naive Bayes algorithm, Pandas and OpenCV libraries used in python languages

David Forsyth- This extraordinary book gives a uniquely modern view of computer vision. Offering a general survey of the whole computer vision enterprise along with sufficient detail for readers to be able to build useful applications, this book is invaluable in providing a strategic overview of computer vision. With extensive use of probabilistic methods-- topics have been selected for their importance, both practically and theoretically--the book gives the most coherent possible synthesis of current views, emphasizing techniques that have been successful in building applications. Readers engaged in computer graphics, robotics, image processing, and imaging in general will find this text an informative reference.

Richard Szeliski- Computer Vision: Algorithms and Applications explores the variety of techniques commonly used to analyze and

interpret images. It also describes challenging real-world applications where vision is being successfully used, both for specialized applications such as medical imaging, and for fun, consumer-level tasks such as image editing and stitching, which students can apply to their own personal photos and videos.

Mark Nixon- Feature Extraction for Image Processing and Computer Vision is an essential guide to the implementation of image processing and computer vision techniques, with tutorial introductions and sample code in MATLAB and Python. Algorithms are presented and fully explained to enable complete understanding of the methods and techniques demonstrated. As one reviewer noted, "The main strength of the proposed book is the link between theory and exemplar code of the algorithms." Essential background theory is carefully explained.

Nusrath Shaikh- In this color detection python project. We are going to build the application in python language and get results to usethe naïve bayes classifiers, pandas and OpenCV libraries areused in this project which can automatically detect the nameof the color. So we have a dataset which have color names and its value. Then we will calculate the distance from each color and find the shortest one. With the advancement of AI Computer vision came into the image in late The 1960s. Its whole purpose was to extend the intellect of the synthetic

mechanism available by installing the cameras into them and describe whatever they saw just like humans' sensory systems.

## **TILL DATE KNOWLEDGE OF OPEN COMPUTER VISION**

Open CV is a programming function library which targets mainly towards the real time computer vision simply talking it is a programming function library whose main function is Image Processing. So mainly speaking it will not do all the task related to the image we need to put the library on our system and start coding which then will use the different features which is present there in the OpenCV. We here do the coding process that is building our code and then run it to do the task we have assigned it. Open CV is a CV library with APIs that permits us to find out a way for our Computer Vision Project. Open-CV-Python is a library of python bindings which is designed for solving problems related to computer vision it also uses Numpy which we know is a highly optimized library for mathematical and numerical operations with a syntax resembling MATLAB. The OpenCV arrays structured are always converted to and from the Numpy arrays.

Using OpenCV and Pandas for color detection

Under this topic I will describe the importance and use of OpenCV and pandas libraries in our project. So first we need to clarify this that Numpy, Pandas & OpenCV are necessary for this project and thus we

need to install them in python.Here we are also using a library to create an argument parser and that is argparse. If we want we can directly read the image in our code by giving the complete image path in the command prompt. For reading the datasets like xlxs csv Pandas library is very important as it reads the dataset using CSV/xlxs.pd.read.csv/xlxs() into the Pandas Data frame.

A name is assigned to each column for accessing it easily. We use the OpenCV library for the detection of name and values of color and also for detecting the image. It does so by calculation of the RGB values of the pixels. There are (X,Y) coordinates of the position where we need to detect color. This function is called the draw function which is created using Pandas and OpenCV. In this paper we use visual C++ & OpenCV for processing and display of image.
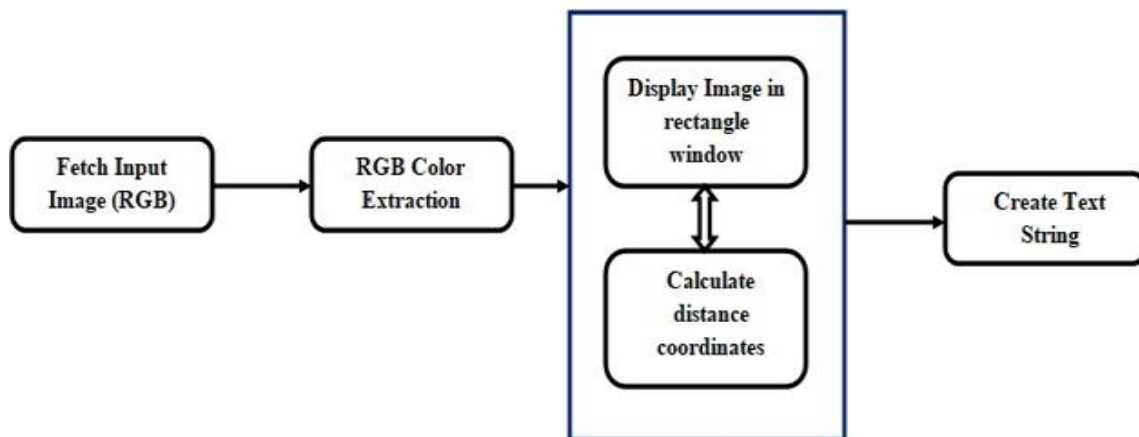
The above architecture shows the capability for the project. It consists of a well defined sequence diagram that is abstracted from the source code. It leverages the rich capabilities of the technology such as OpenCv

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.

Chapter-3

Functionality/Working of Project

Image Capturing:- First and the foremost step is to extract a image with high resolution.To do so we run the read image command

```
path = r'C:\Users\Welcome\Desktop\test photo.jpg'

# Using cv2.imread() method
img = cv2.imread(path)

# Displaying the image
cv2.imshow('image',img)
```

Extracting the RGB(Red, green , blue) colors:-

During this step from the input image we fetch 3 layered colors. Whatever color we see on any screen including our PC, laptop , mobile phones , TV all are a combination of there basic three RGB colors. So the lowest value that a primary color takes is 0 and it extends upto 255 ie the highest. When we mix all the 3 basic colors that are red, green and blue a huge wide range of colors are produced. If I explain with an example , if we get the intensity value of our primary color 0 then the corresponding color is black and if the intensity of our primary color is 1 then the corresponding color is white

```
#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)
```

Calculating the minimum distance from the x,y coordinates:-

We calculate the minimum distance by going towards the origin (0,0) from the huge range of all the colors to get the mostly same or nearly matching color.

For loading the comma separated values into the pandas Data frame, the Panda library plays a vital role

```python
csv = pd.read_csv('colors.csv',names=index, header=None)
```

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

We use the rectangle window to display images with different shades .We do a double click after that the RGB values associated to that color and name of the color is displayed

```python
#cv2.rectangle(image, startpoint, endpoint, color, thickness)-1 fills entire rectangle
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)
```

The above functions are used to get the intensity level of the color and also its name

STEP WISE STEPS IN BRIEF ARE:-

1. Download and unzip the zip file

2. Taking an image from the user

3. we read the CSV file with pandas

4. Create the draw_function

5. Calculate distance to get color name

6. Display image on the window

7. Run Python File

1. Download and unzip the zip file

Color Detection Zip File

The project folder contains 3 files:

- **Color_detection.py** – main source code of our project.
- **Colorpic.jpg** – sample image for experimenting.
- **Colors.csv** – a file that contains our dataset.

2. Taking an image from the user

We are using argparse library to create an argument parser. We can directly give an image path from the command prompt:

```
import cv2
import numpy as np
import pandas as pdimg_path = "D://OpenCV//shape-
detection//New folder//color palette.jpg"
img = cv2.imread(img_path)
```

3. Next, we read the CSV file with pandas

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.

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

4. Set a mouse callback event on a window

First, 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.

```
cv2.namedWindow('color detection')
cv2.setMouseCallback('color
detection',draw_function)
```

5. 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.

```
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)
```

6. 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.

```
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
```

Our distance is calculated by this formula:

$d = abs(Red − ithRedColor) + (Green − ithGreenColor) + (Blue − ithBlueColor)$

7. 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 the window using **cv2.rectangle** and **cv2.putText()** functions.
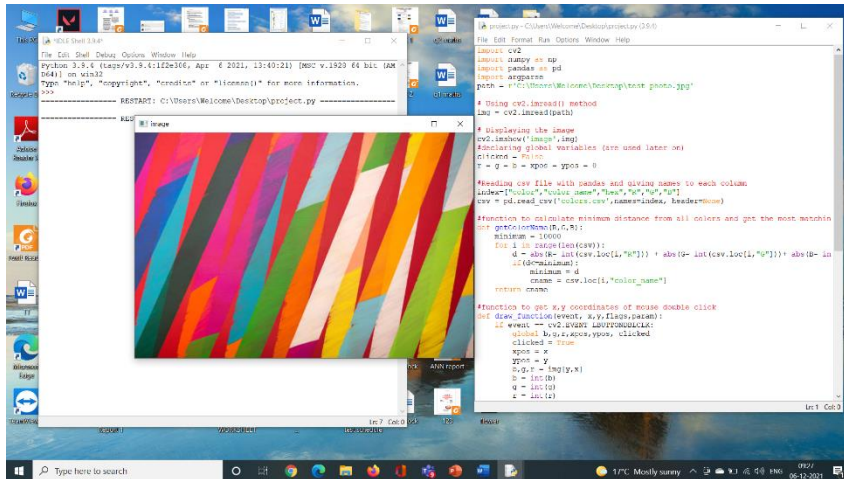
```
while(1):cv2.imshow("color detection",img)
    if (clicked):
        cv2.rectangle(img,(20,20), (750,60),
(b,g,r), -1)         text =
getColorName(r,g,b)+'R='+str(r)+'G='+ str(g)
+'B='+ str(b)
        cv2.putText(img, text,(50,50),2,0.8,
(255,255,255),2,cv2.LINE_AA)
if(r+g+b>=600):
            cv2.putText(img,
text,(50,50),2,0.8,(0,0,0),2,cv2.LINE_AA)

        clicked=Falseif cv2.waitKey(20) & 0xFF
==27:
        break
cv2.destroyAllWindows()
```

Chapter-4

Results and Discussion

EXPERIMENTAL RESULTS

Normal Output

Here the intensity and color name is shown ie papaya whip and
R=255 G=227 B=213

Here the intensity and color name is shown ie fire engine red and
R=193 G=22 B=41

Chapter 5

Conclusion and Future Work

CONCLUSION

So the main motive of this paper was to get the needed color from the set of basic colors ie RGB colors. Here all the steps are implemented using pandas numpy and open CV. The best part of this implementation was differentiation of mono colors. In our future scope the technique can further be extended to detection of color from playing video and many other wide applications.

Computer vision can be used to solve the most intriguing problems with utmost sophistication. All the basics regarding the detection technique along with different ways to achieve it have been profoundly discussed. During the course of programming, we can use both Python and MATLAB for Computer Vision, but we prefer Python because it takes less simulation time than MATLAB. Someone having prior coding experience finds it easy to implement. Contours, shapes & colors were detected in the given sample images successfully.

FUTURE WORK

In the current time there is no accurate representation of colors with full accuracy. In this project we have used the computer vision dataset and accordingly the color is identified using RGB and hex values of the

image. In coming future we will also be able to detect color from the playing video.

Computer vision has still not attained a level wherein it can be directly put into use to solve life problems, as it is still in its developmental phase. With passing years and rigorous pace at which research is being done, Computer Vision or to be precise, Object detection will be completely omnipresent. Computer Vision is a sub-part on Machine Learning. Some common and widely used application of object detection are:

6.1 Face Recognition Have you ever thought how iPhone X recognizes your face when you unlock it? Not only it detects, it remembers the face too for next time for the phone to be unlocked. This is one of the simple applications of face recognition that we can see in our daily life.

6.2 Accounting Number of Objects Computer Vision is also used for counting purpose.For example, when you upload a photo on Facebook, it tells/counts the number of people in that photograph.

6.3 Automobile Spotting When an object to be detected is a vehicle, it's automobile spotting. There is live tracking on moving Traffic. It is already used on highways and traffic lights for detection of car plate who violates traffic rules.

Reference

[1] Reetu Awasthiand Khushboo Khurana, "Techniques for Object Recognition in Images and Multi-Object Detection",(IJARCET), ISSN:2278-1323,4th , April 2013.

[2] Artificial Intelligence. (December 2017). Retrieved from https://en.wikipedia.org/wiki/Artificial_intelligence

[3] Computer Vision. (December 2017). Retrievedfrom https://en.wikipedia.org/wiki/Computer_vision

[4] Computer Vision's Open Source. (December 2017). Retrieved from https://en.wikipedia.org/wiki/OpenCV

[5] (2017, December 17). Object Detection [Online]. Available: http://en.m.wikipedia.org/wiki/Object_detection

[6] Numpy.org, 2017. [Online]. Available: http://www.numpy.org

[7] https://www.invensis.net/blog/it/benefits-of-python-overother-programming-languages/

[8] Rosebrock, A., 2020. Opencv And Python Color Detection - Pyimagesearch. [online] PyImageSearch. Available J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[9] vkisku/Face-Detection-OpenCV. GitHub. (2020). Retrieved 23 June 2020, from https://github.com/vkisku/Face-DetectionOpenCV.

[10] RGB Color Codes Chart Rapidtables.com. (2020). Retrieved 23 June 2020, from https://www.rapidtables.com/web/color/RGB_Color.ht ml.

[11] Rosebrock, A. (2020). Measuring distance between objects in an image with OpenCV - PyImageSearch. PyImageSearch. Retrieved 23 June 2020, from https://www.pyimagesearch.com/2016/04/04/measuringdistance-between-objects-in-an-image-with-opencv/.

[12] Zelinsky, A. (2009). Learning OpenCV---Computer Vision with the OpenCV Library (Bradski, G.R. et al.; 2008)[On the Shelf]. IEEE Robotics & Automation Magazine, 16(3), 100- 100. https://doi.org/10.1109/mra.2009.933612.

[13] Naive Bayes Classifier in Machine Learning - Javatpoint. www.javatpoint.com. (2020). Retrieved 23 June 2020.

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