Color Detection

Subham Das
IISER Bhopal
EECS Department
subham18@iiserb.ac.in

***Abstract*—Color detection is the process of detecting and naming the color. This is an easy task for humans, but computers cannot detect color easily. Using Naive Bayes algorithm, Pandas, and Open-CV libraries of python language, we can create a program to mimic this function. Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the category labels are drawn from some finite set. Naive Bayes classifiers assume that the worth of a specific feature is independent of the worth of other features, given the category variable. Open-CV was designed for computational efficiency and specializes in real-time applications. Pandas is a cloud-based platform that gives video and audio encoding infrastructure. We take input from the device camera and process pixel values to produce the color using HSL color space. The detected color will be displayed as a label on the screen.**

***Index Terms*—HSV, contour, open-CV, pixels, color**

I. Introduction

Three different colors Red, Green, and Blue, are being tracked by utilizing the fundamentals of computer vision. We have employed two different methods.

The first one is color detection using Open-CV and python to detect color with the inbuilt webcam of computers/laptops. This recognizes Red, Green, and Blue using the live-feed visual from the webcam to give real-time identification of colors present on the screen.
Open-CV is Open Computer Vision Library. It enables real-time computer vision. This library can easily interface with programming languages like Python, MATLAB, and others as well. Along with Numpy and Pandas, image processing can be performed at ease.

The second method is identifying all shades of color (as hex codes) from an image. The three components, **hue, saturation, and lightness**, are independent in HSL color space, so are more suitable for color image analysis than RGB color space. We convert the pixel value from RGB to HSL before determining the color and then map the global coordinates to the widget's local coordinates. Any transform applied to the widget (like scaling it down to fit the screen) will modify the pixel coordinates.
In the case where image bytes are taken directly from the asset, we need to map pointer (touch/mouse) coordinates to the image size coordinates with high precision and accuracy.

TABLE I
HSV Range

Hue Range	***Color Category***	***Hue Range***	***Color Category***
0-15, 346-360	Red	136-160	Spring Green
16-40	Orange	161-200	Cyan
41-45	Gold	201-255	Blue
46-60	Yellow	256-300	Purple
61-90	Yellow-Green	301-345	Pink
91-135	Green		

One of the applications of color detection by computer vision is in driver-less cars. This system is helpful in detecting traffic and vehicle back-lights and takes a decision to stop, start and continue driving. The project has been extended to use the capabilities of color detection to calculate the heart rate of individuals.

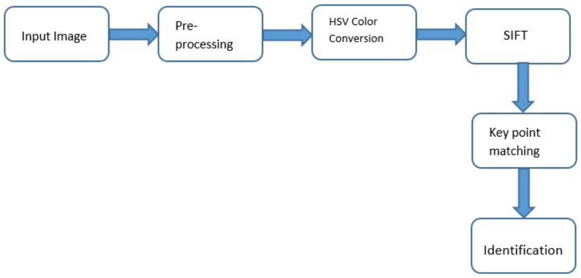


Fig. 1. The Process

II. Implementation

Using Python - The first step is to fetch a high-quality image with resolution (done through webcam). To load an image from a file, we use cv2.VideoCapture(). Each primary color takes an intensive value 0 (lowest) to 255 (highest). When mixing three primary colors at different intensity levels, a variety of colors are produced. For example, if the primary colors' intensity value is 0, this linear combination corresponds to black. If the intensity value of the primary colors is 1, this linear combination corresponds to white.
The Pandas library serves as an important utility to perform various operations on comma-separated values like pd.read_csv() reads the csv file and loads it into the Pandas data frame.
The above architecture makes the process more efficient based on principles and properties related to each other. 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.

Using the Open-CV library we are able to get accurate contours around objects before labeling them a color

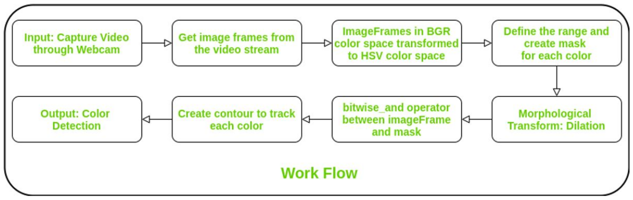

Input: Capture Video through Webcam
Get image frames from the video stream
ImageFrames in BGR color space transformed to HSV color space
Define the range and create mask for each color
Morphological Transform: Dilation
bitwise_and operator between imageFrame and mask
Create contour to track each color
Output: Color Detection
Work Flow

Fig. 2. Python and Open-CV

Using Android Studio - This method involves breaking down the image into many pixels. We then convert to ByteData with rawRgba format knowing the width/height of the image. The bytes are then stored as a list, and we loop through it till it matches with the color library. Each value in the list is a component for the color channel (from 0-255), list[0] red pixel 1, list[1] green pixel 1, list[2] blue pixel 1, list[3] alpha pixel 1, so each 4 values u can make a Color(). Basically, we have made a method for getPixel(x,y) and then calculated what would be the index in that cartesian coordinate.

The color-matched is then displayed in the top left corner as the hex code (AARRGGBB).

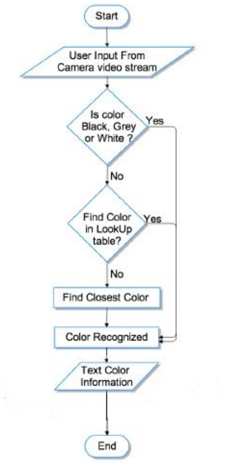
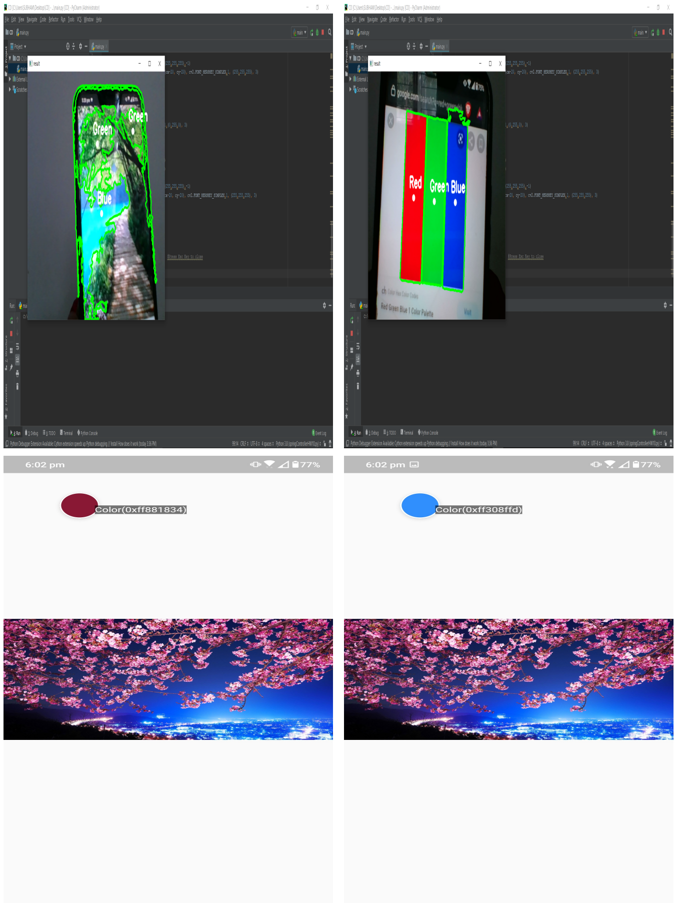

Start
User Input From Camera video stream
Is color Black, Grey or White ?
Yes
No
Find Color in LookUp table?
Yes
No
Find Closest Color
Color Recognized
Text Color Information
End

Fig. 3. Logic Flow

III. Results and Future Work

We have identified a number of shades along with their RGB and hex values. Whenever the cursor clicks the image, it automatically shows the RGB color values. The heart rate monitor app was developed using the methods and ideas of this project as its base, and it may also have several other biomedical applications. The RGB detector using open-CV and python may have suitable applications in object detection, facial recognition, etc.



Color(0xff881834)

Color(0xff308ffd)

Fig. 4. Output from the Different Methods

IV. Conclusions

In this paper, we successfully got the required color field (RGB) from an image and video. The Open-CV method allowed us to identify different shades of RBG and group them within the three while identifying the color class from a live video feed. The color detector app successfully gives the hex code of any color on an image. The future scope of this is edge detection techniques that has other applications like facial detection, color conversion for greyscale image, etc. all this can be implemented from the above method.

References

[1] Doweesh, Al-Hamed, Al-Khalifa, ”A Real-time Color Identification Mobile Application for Visually Impaired People”, Springer International, 2014

[2] Puri, Gupta, Sikri, ”CONTOUR, SHAPE, AND COLOR DETECTION USING OPEN CV – PYTHON”, International Journal of Advances in Electronics and Computer Science, 2018

[3] Raguraman, Meghana, Navya, Karishma, Iswarya, ”Color Detection of RGB Images Using Python and OpenCv”, International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2021

[4] Farooqi, Chauhan, Siddiqui, ”Naive Bayes Approached in Color Detection using Pandas and OpenCV”, International Journal of Scientific and Engineering Research, 2020