Ariana LaCue

March 18, 2020

CPS 3320

Color Detection using Pandas & OpenCV

Background:

How many times has it occurred to you that even after seeing, you don’t remember the name of the color? Evidently, there can be 16 million colors based on the different RGB color values, but we only remember a few. In this project, I will build an interactive program that will detect the selected color from any image. The colors being detected will have a sample given binary images using Python 3.7, 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.

Hypothesis:

People who are color blind will benefit from this program. The program will allow the detection of a specific color in a livestream video content. A video is composed of infinite frames at different time instants. We will detect the color of every frame one by one.

Data:

In this color detection Python project, I built an application through which I can automatically get the name of the color by clicking on them. For this, I have the data file that contains the color name and its values. Afterwards, I calculate the distance from each color and find the shortest one.

Colors are made up of 3 primary colors; red, green, and blue. In computers, I can define each color value within a range of 0 to 255. There are approximately 16.5 million different ways to represent a color. In the dataset, I need to map each color’s values with their corresponding names. I will be using a dataset that contains RGB values with their corresponding names. The CSV file for our dataset has been taken from colors.csv. The colors.csv file includes 865 color names along with their RGB and hex values.

Analysis Steps:

First. I take an image from the user. I used argparse library to create an argument parser. I directly gave an image path from the command prompt. Next, I read the CSV file with panadas. The pandas library is very useful library, I perform various operations on data files like CSV. pd.read\_csv() reads the CSV file and loads it into the pandas DataFrame. I assigned each column with a name for easy access. Afterwards, I set a mouse callback event on a window. Then, I created a window that will display the input image I will then set a callback which will be called a mouse event. The callback function will be called the draw\_function() whenever a mouse event occurs. Next, I created a draw function. It will calculate the rgb values of the pixel. The function parameters have the event name, (x,y) coordinates of the mouse position, etc. Continuing, I calculate the distance to get the color name. Finally, I display the 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. When the user double clicks the window, the program will draw a rectangle and get the color name to draw text on the window using cv2.rectangle and cv2.putText() functions.

Challenges:

The main purpose is to detect various colors in a given sample image. The most challenging part in this task was to determine a color is a shadowy area. Shadows made the color darker than it already should be. Also, I had an error that would say “Traceback (most recent call last): File “color\_detection.py”, line 1, in import cv2 ModuleNotFoundError: No module named ‘cv2’”. I found out that my python couldn’t find the module that I was importing. I had to download the OpenCV module, and type this in the command line:”pip install opencv-python”. Afterwards, it was working fine.

Results:

In conclusion, Computer vision (OpenCV) can be used to solve the most intriguing problems with the utmost sophistication. All the basics regarding the detection technique along with different ways to achieve it have been profoundly discussed.

Reflection:

In this Python project I learned about colors and how to extract color RGB values and the color name of a pixel. I have a better understanding of how to handle events like double-clicking on the window and how to read CSV files with pandas and perform operations on data. This is used in numerous image-editing and drawing apps. If I do this project again, I would want to figure out how to detect the color in dark lighting.

All in all, I’m glad that what I’ve learned went beyond the knowledge to this project. I learned that if I want to pick up a new skill or improve a skill, it is about having the commitment and putting in the hours to learn and practice. There are rarely any shortcuts. I have accepted that I learn faster when I’m working on a project, I’m passionate about. As I was really keen to build a website for my triathlon club, I picked up Ruby on Rails much faster than when I tried to learn Python by doing tutorials. It also helped that I was focusing on a few priorities. I learned how to search for answers myself. It would be great to have someone to help me whenever I’m stuck, but I might not be able to get help all the time. Thanks to Google and the nice code samurai at Kean who made countless resources and guides online, finding a possible resolution to a problem has become much easier. I learnt that I might not always find the best solution immediately, but I can implement whatever works for now and iterate when I find a better solution or gain more knowledge on the topic.