BLG453E Homework-1 Report

1 - Part 1: "I am feeling blue"

In this part of the assignment, I started by reading the frames of a video and the color-separated versions of the objects in the given data set. I started this reading process with the help of "glob" library, first reading the file names and then assigning the images to a list I named "image_list". My list, which I call "image_list", represents the frames of the video, and the objects identified in the "segmented_image_list" list that I have filled in with the same method.

After I finished reading the file, I started navigating the image pixels. In Grayscale mode, when I was browsing pixelated pixels on the segmented images I read, I set myself to 38. After assigning the coordinate to which I captured the value 38, I reduced the value in the red and green channels by 75 percent to give the blue hue of this value in the actual video frame as given in the assignment description. I then processed it on all video frames and printed it with the help of the "movie.py" library.

The part I had difficulty with in this part of the assignment was that the opencv library read the images in BGR mode, not RGB, and navigated the images using the for loop. I was able to compare it with a vectorized process shown in the assignment description, and realize that it runs much, much slower with the for loop...

Outputs like that:

2 - Part 2: Histogram matching

In this assignment, I started doing file readings as in part1. Then I found the target image for the histogram mapping, the image named "Dinasour JR" given in the assignment text.

I created the histogram matching function, "hist_match" by performing the following operations and getting help from the web [1].

- Add new axis to images without shape 3
- Create for loop for every 3 color channel of the picture
- Assigning histograms of images to the function via numpy
- Finding cumulative sums
- Making normalizations
- Calculate interpolations
- Reflect the result to a picture

In the explanation of the assignment, I applied the following procedures for the part which was presented as 2 options and we could not give full meaning.

- I concatanete all of the video frames and I get a single picture
- Then I matched the histogram of this picture and the histogram of the target picture and printed it on the incoming video frame.

Finally, as in part1, I list the pictures and print them on video with a value of 25 fps.

Outputs like that :

3 - Part 3: Histogram matching from segmentation maps

In this part of the assignment, a combination of the operations we do in part1 and part2 is desired. For this process, I read the file in the previous sections.

After performing file reading operations, I separated the objects in each image as a separate image with the help of segmented images. I assigned each object in the image as a separate image, and in each of these images, I selected the blue, red, purple, and yellow weighted images I placed in the "target_images" folder and performed a histogram match.

After applying the histogram mapping event to each object, that is, to each object, I assigned these pictures to the variable I named "result" and merged those pictures.

As a result of this process I get an output as follows :
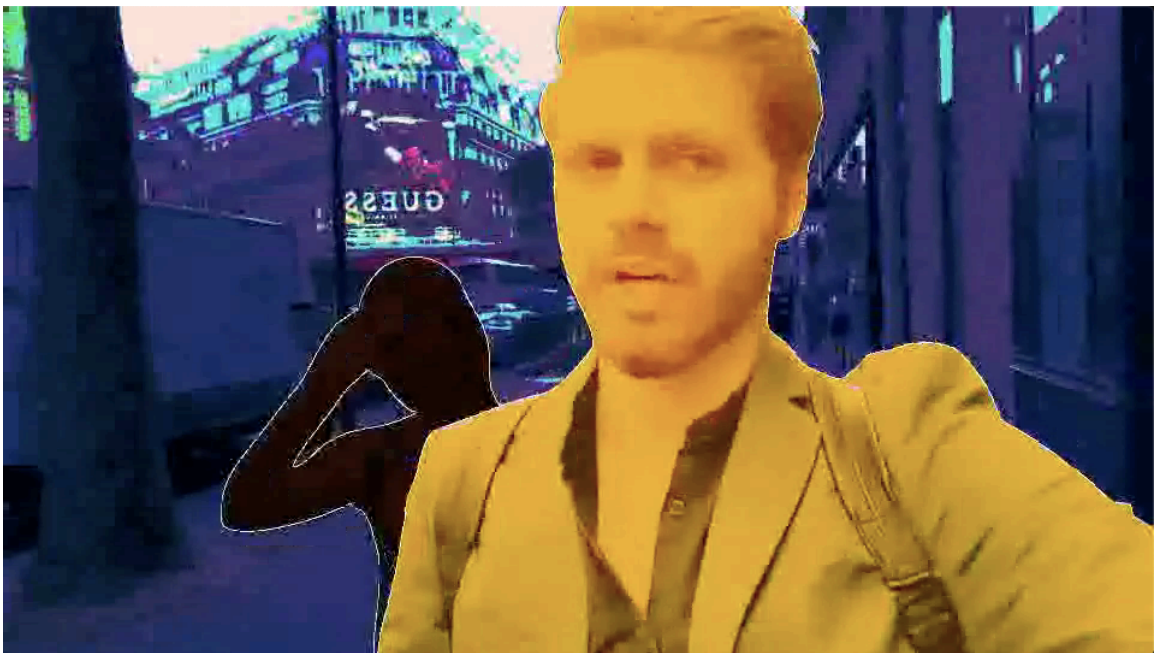
In this illustration, the following histograms are applied according to the segmentation color.

- I applied a yellow histogram if the segmentation is red
- I applied a red histogram if the segmentation is green
- I applied a purple histogram if the segmentation is black

Okan Dönmez / 150170708

[1] https://vzaguskin.github.io/histmatching1/