

# ROBT 310 Image Processing - Project 4 Report

Anuar Maratkhan

## I. INTRODUCTION

This is the report for ROBT 310 project 4. The project is aimed to evaluate students' understanding of selective coloring effect. Student are required to complete the task using OpenCV (Python was chosen). Below I will describe the way I tackled the problem of selective coloring effect and my bonus feature.

## II. WHAT HAS BEEN DONE

Firstly, I have translated the code written in MATLAB that was taken from inclass exercise to Python written with OpenCV and NumPy. The image has been also chosen the same as in MATLAB inclass exercise, the 'parrot.jpg' due to its colorfulness.

Then, next point was to add trackbar for changing radius. That was tough enough because I was new to developing GUI in Python. However, the problem has been solved, and the radius taken from added trackbar was passed to a function that makes selective color effect from an image (`color_slice`). The problem was that even if the radius does not change at the instant time, the function is called again and again. That cause me waste of hardware resources because I was computing the same thing so many times. Therefore, the way I have solved the problem is by adding `prev_rad` variable to track previous radius used in selective coloring. Then, compared it with current, and if it has not changed, I will not compute it again. That solved me a problem.

Furthermore, I decided to add mouse callback. Thanks to good documentation of OpenCV (even though most of it for C++), I could handle that easily. However, one thing made the problem a bit harder. In such images we need to interchange X-axis and Y-axis in pixel positions. That caused me a little trouble which I have solved then.

After that, I need to add camera to the project. Due to my misunderstanding the project description, I have started finding ways to add external camera connected through USB, which I have connect through USB with my iPhone. That was unsuccessful for me and I could not connect iPhone camera. I even had an idea to find separate web-camera and connect it with USB. Thanks to Olzhas Adiyatov for clarifying that we needed to connect any camera. Moreover, connecting camera was not hard for me. But after connecting external cam I have found that my code runs so slow... So that I thought of rewriting everything in C++. Hopefully, I understood that I was using pythonic for loops for computing the selective color on NumPy arrays! What a mess... I then, started manipulating those arrays, first, using `nditer` that returned me an iterator. However, after playing with it, I understand that it is not the solution, and I have stopped on more efficient solution that included manipulation with whole arrays and matrix operations. As far as I know, such NumPy matrix operations

are written in Fortran and are fast enough. And it was, actually, because I have reduced computation time from 3.5 seconds down to less than half a second. Before resizing my image, it took me more, to tell the truth. So, by the end of computation, I had a mask of `True` and `False` values which indicated whether my computation for a color slicing enclosed by a **sphere** is more than certain selected radius or not. After, I have applied that mask to make not selected colors gray (mean).

Finally, I started trying to add bonus feature which was not that successful for me. I wanted to implement something like filter in here. However, I have discovered that OpenCV does not directly convert RGB to CMYK colormap, so I have used `YCR_CB`. After that I wanted to make shift effect, and firstly was trying to do so using affine transformations. However, that it was unsuccessful and I have found NumPy's `roll` function which I have used as shift along horizontal axis.

## III. HOW TO USE

Just type `python3 main.py` in your terminal window (project directory). Firstly, the program will show a parrot image, and you first need to press 'Esc' (escape) to close the window, and to make web camera open. Again, if you want to close the window, press 'Esc'. The program will stop. For choosing the color for color slicing, just press the region in the image with certain color.