HOMEWORK ASSIGNMENT

NAZARBAYEV UNIVERSITY | SCHOOL OF SCIENCE AND TECHNOLOGY

PROJECT 4

In this project we will be working on creating a Selective Coloring effect. The assignment should be completed using OpenCV to fortify the knowledge by practice.

DUE DATE

Friday, 30th of March

METHOD OF DELIVERY

Assignment deliverables should be submitted via Moodle to the course instructor before the due date.

LEVEL OF COLLABORATION ALLOWED

Discussing the project and helping each other for setting up OpenCV are allowed for this project. In your report, indicate the students you have collaborated with. However, each student should complete and submit their own assignment. The bonus feature button should be unique for each student.

ESTIMATED TIME FOR COMPLETION

20 hours

ADDITIONAL SUPPORT

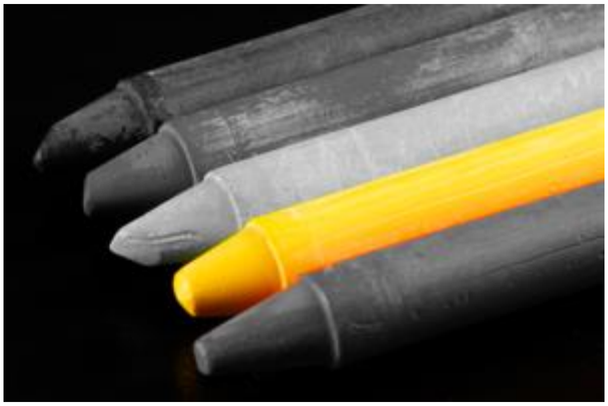
Please contact the course instructor if you need any assistance or have any concerns about this assignment.

GRADING CRITERIA

- 60% will go for the task accomplishment, correct application of image processing concepts, mathematical tools, and OpenCV programming skills.
- 10% of the assignment will go for the proper documentation of the work (report).
- 20% are awarded if you incorporate a USB camera as a source for input images and perform the operation continuously in real-time.
- 10% are given for a “bonus feature button” of your program – the grade up to 10 will be generated from peer evaluation – students will rank each other’s surprise buttons.

ASSIGNMENT DETAILS

In this assignment, you will be working on selective coloring effect. This effect is usually used in tracking applications (finding the red object in a scene and tracking it) or in movies. The aim of selective coloring, as you already know from lectures, is highlighting a specific range of colors , while keeping the background in grayscale. See the pictures below as an example.



For this we use regions defined by the colors. You are familiar with two cases:

1. Colors of interest enclosed by a **cube**.

$$s_i = \begin{cases} 0.5 & if \ \left[\left|r_j - a_j\right| > \frac{W}{2}\right]_{any\ 1 \le j \le n} \\ r_i & otherwise \end{cases} \quad , i = 1, 2, \dots, n$$

2. Colors of interest enclosed by a **sphere**.

$$s_i = \begin{cases} 0.5 & if \sum_{j=1}^{n} (r_j - a_j)^2 > R_0^2 \\ r_i & otherwise \end{cases} \quad , i = 1, 2, \dots, n$$

Your task in this assignment is to utilize your knowledge of selective coloring and implement it in a program written using OpenCV. The program should allow a user to direct the cursor on an image and obtain the RGB value of a pixel by a clicking on it. You then need to enclose the color by a sphere (or a cube) and be able to increase/decrease the edge length or the radius using a trackbar (have some default value pre-set greater than 0). The effect of changes caused by manipulating the trackbar should be immediately seen on the target image in the display section.

As you have seen from the Grading Criteria section, 20% of the grade will be awarded for successful integration of a USB camera into the system. Basically you are required to connect the camera to your program and enable a continuous stream of frames at a given frame rate (camera's default). Program should then process every single frame incoming from the camera and show the resulting image to the display area of the program's GUI.

Finally, there should be a "bonus feature button" on your GUI. It is up to you how to use it – the aim of this button is to stimulate creativeness. Therefore you need to think of what feature would the button enable which might surpise and impress your groupmates, as it is them who will grade it.

Following links will help you to start with video capture class and GUI:

1. http://docs.opencv.org/modules/highgui/doc/reading_and_writing_images_and_video.html#videocapture
2. http://docs.opencv.org/doc/tutorials/highgui/table_of_content_highgui/table_of_content_highgui.html#table-of-content-highgui

INSTALLATION INSTRUCTIONS

This year you have two options for programming the project: C++ or Python. Here are installation video guides for Python with OpenCV:

Windows 7: https://goo.gl/H1nuAw
Ubuntu 16.04: https://goo.gl/ViVQB7

You can use Jupyter notebook for prototyping and VScode (https://code.visualstudio.com) for programming an actual program. There is a good tutorial on programming on Python in VScode, follow the link https://code.visualstudio.com/docs/python/python-tutorial .

For C++ version for Windows please watch the following video:
https://www.youtube.com/watch?v=l4372qtZ4dc
and for C++ version on Ubuntu please read this tutorial:
https://gist.github.com/olzhas/575a3737e54245c9f050e8fc7c06db2a

TA Olzhas Adiyatov will provide help and guidance for the installation process and also for C++ programming.