VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY

USER GUIDE
Individual Lab 1

Computer Vision

Performed by:
Student's ID: 21127730
Student's name: Hoang Le Cat Thanh

Instructors:
Pham Minh Hoang

Ho Chi Minh City, March 4, 2024

Contents

1 Introduction

The project is the implementation of several image manipulations using C++ including:

- Convert a color image to the gray image
- Change the brightness/contrast of an image
- Filter an image using Average, Median and Gaussian filter
- Edge detection using Sobel and Laplacian filter

2 Installation and set up

- Install OpenCV library
- Set up environment variables
- Configure project
  - Open project's *Properties*, add *Include Directories* and *Library Directories* as well as lib file.
  - Copy *opencv world341d.dll* to the Debug folder containing the exe file.

3 Running program

3.1 Command line

3.1.1 Convert color image to grayscale

⟨*Executable file*⟩ -rgb2gray ⟨InputFilePath⟩ ⟨OutputFilePath⟩

where

- *-rgb2gray*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image

3.1.2 Change image brightness

⟨*Executable file*⟩ -brightness ⟨InputFilePath⟩ ⟨OutputFilePath⟩ ⟨C⟩

where

- *-brightness*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image
- *C*: the brightness factor (integer). Postive value will result in brighter image while negative value will lead to darker image.

3.1.3 Change image contrast

⟨*Executable file*⟩ -contrast ⟨InputFilePath⟩ ⟨OutputFilePath⟩ ⟨C⟩

where

- -*contrast*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image
- *C*: the contrast factor (float). Value greater than 1 will enhance image contrast while value between 0 and 1 will lower image's contrast.

3.1.4 Filter image using Average filter

⟨*Executable file*⟩ -avg ⟨InputFilePath⟩ ⟨OutputFilePath⟩ ⟨k⟩

where

- -*avg*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image
- *k*: the filter kernel size (integer). The program only takes in odd value from 3 as valid kernel size. Larger kernels result in more extensive averaging or blurring of pixel values, leading to a smoother appearance in the image.

3.1.5 Filter image using Median filter

⟨*Executable file*⟩ -med ⟨InputFilePath⟩ ⟨OutputFilePath⟩ ⟨k⟩

where

- -*med*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image
- *k*: the filter kernel size (integer). The program only takes in odd value from 3 as valid kernel size. Larger kernels result in more extensive averaging or blurring of pixel values, leading to a smoother appearance in the image.

3.1.6 Filter image using Gaussian filter

⟨*Executable file*⟩ -gau ⟨InputFilePath⟩ ⟨OutputFilePath⟩ ⟨k⟩

where

- *-gau*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image
- *k*: the filter kernel size (integer). The program only takes in odd value from 3 as valid kernel size. Larger kernels result in more extensive averaging or blurring of pixel values, leading to a smoother appearance in the image.

3.1.7 Image edge detection using 3x3 Sobel kernel

⟨*Executable file*⟩ -sobel ⟨InputFilePath⟩ ⟨OutputFilePath⟩

where

- *-sobel*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image

3.1.8 Image edge detection using 3x3 Laplace kernel

⟨*Executable file*⟩ -laplace ⟨InputFilePath⟩ ⟨OutputFilePath⟩

where

- *-laplace*: command for the function
- *InputFilePath*: the path of input file/ image
- *OutputFilePath*: the path of output file/ image

3.2 Call the program

- Run the program once to create exe file. (Assuming that the *opencv world341d.dll* has been copied to the project)
- The executable file is located in the debug file after the project run once.
- In the *Debug* folder, right click and choose *Open in Terminal*.
- After open the Terminal, enter the desired command line from the list of given command line with the *Executable file* is named *21127730*.
- If there is no error, the input and output image will be shown. After that, press Enter to close the images and save the output. If save successfully, there will be a notification on where the image is saved.

3.3 Sample process

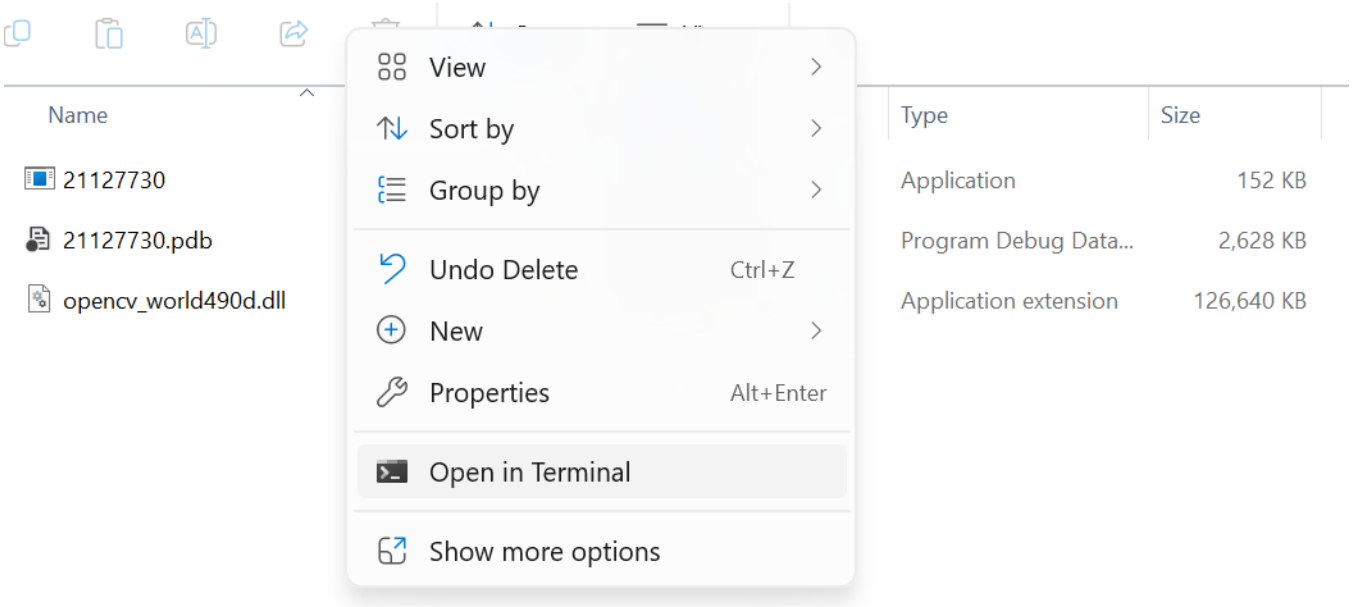

Figure 1: Open Terminal

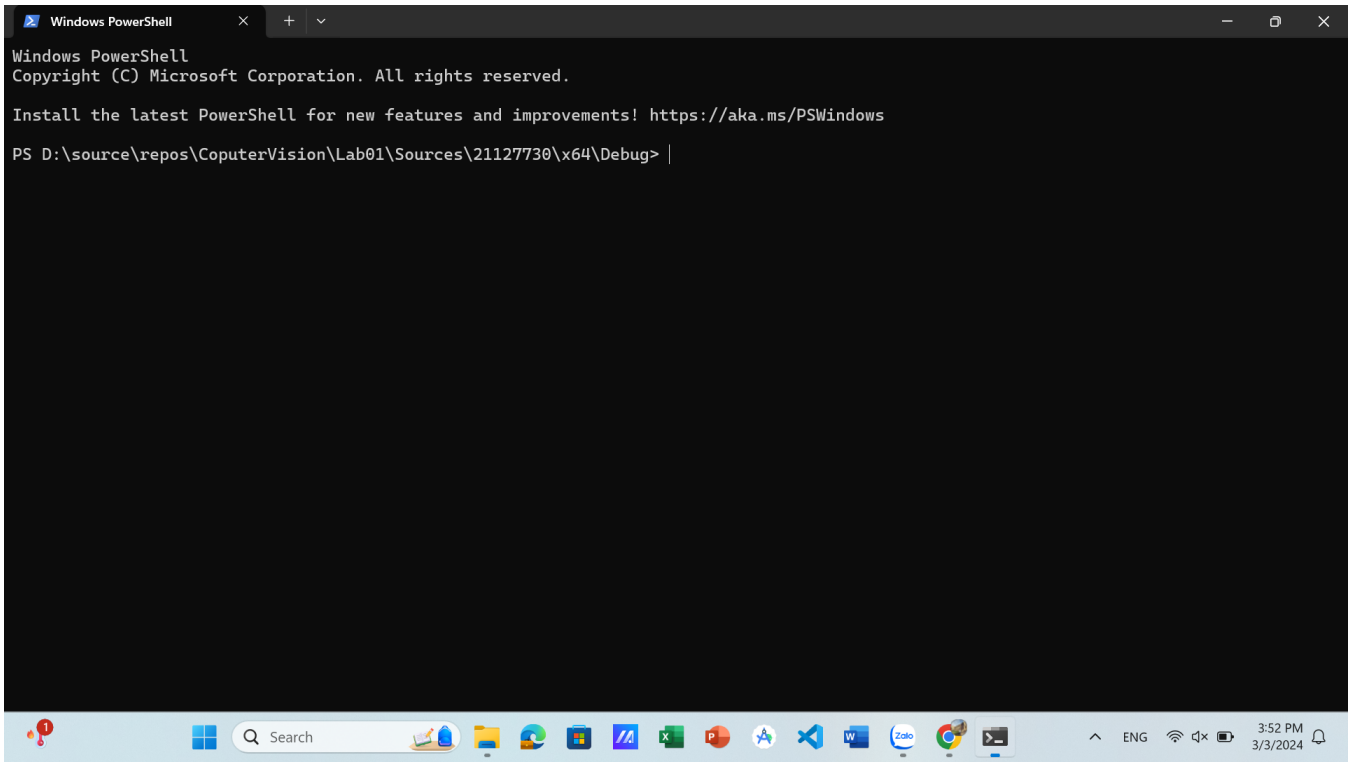

Figure 2: Terminal screen

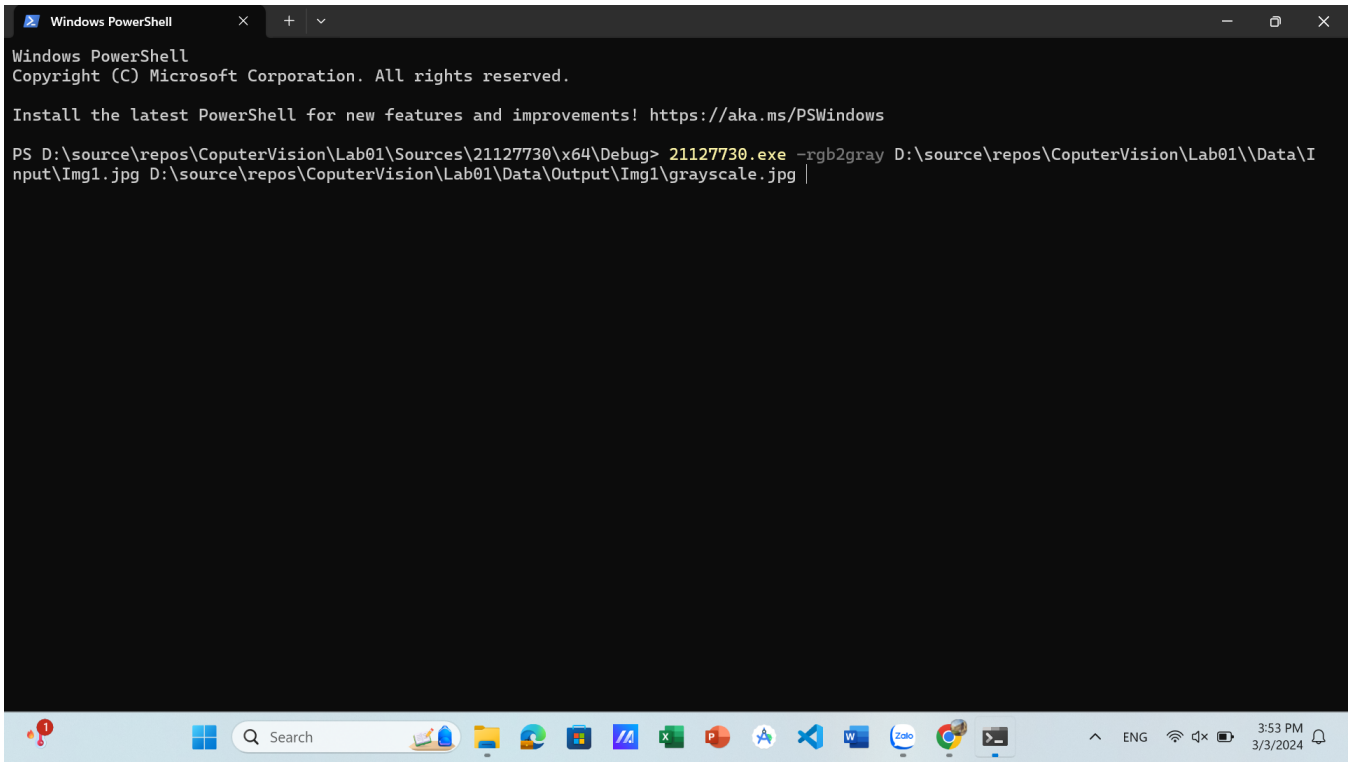

Figure 3: Enter the command line (convert to grayscale image)

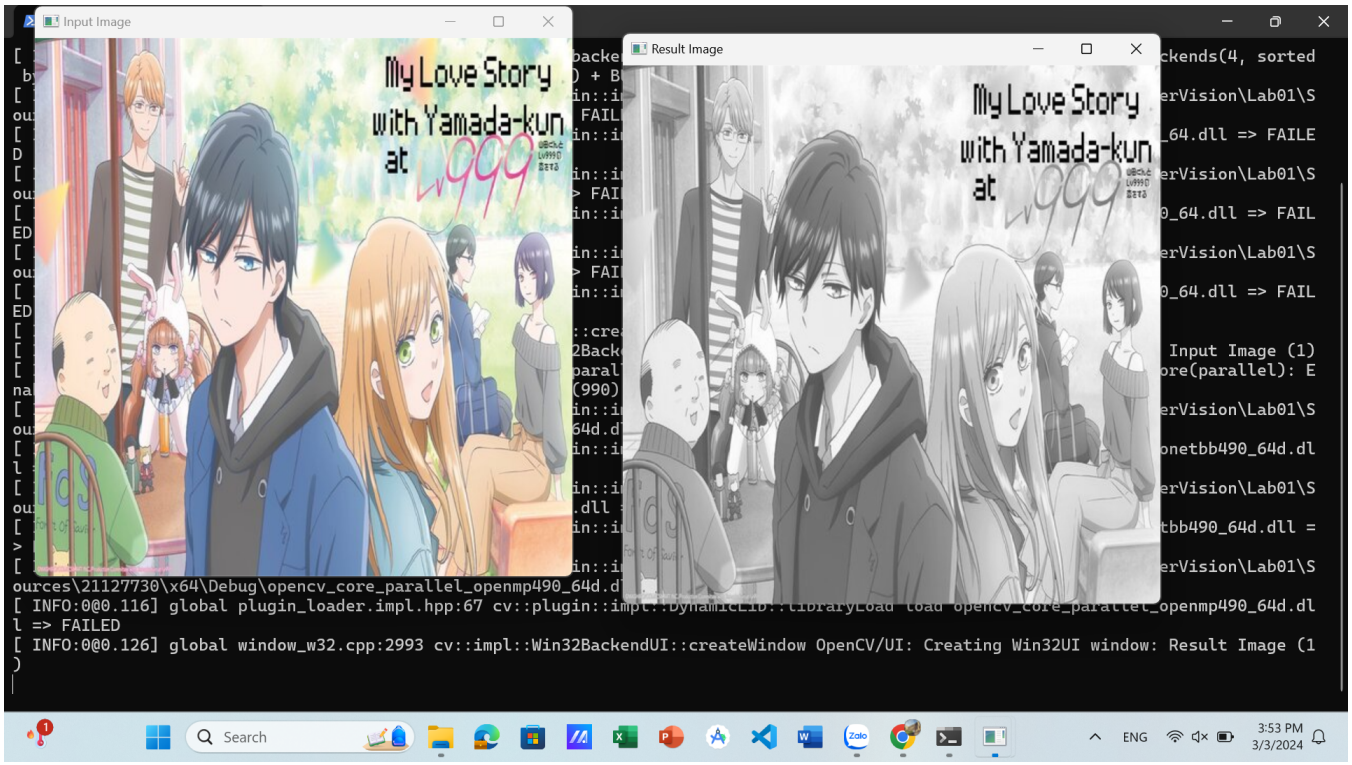

Figure 4: Input and Output image after process

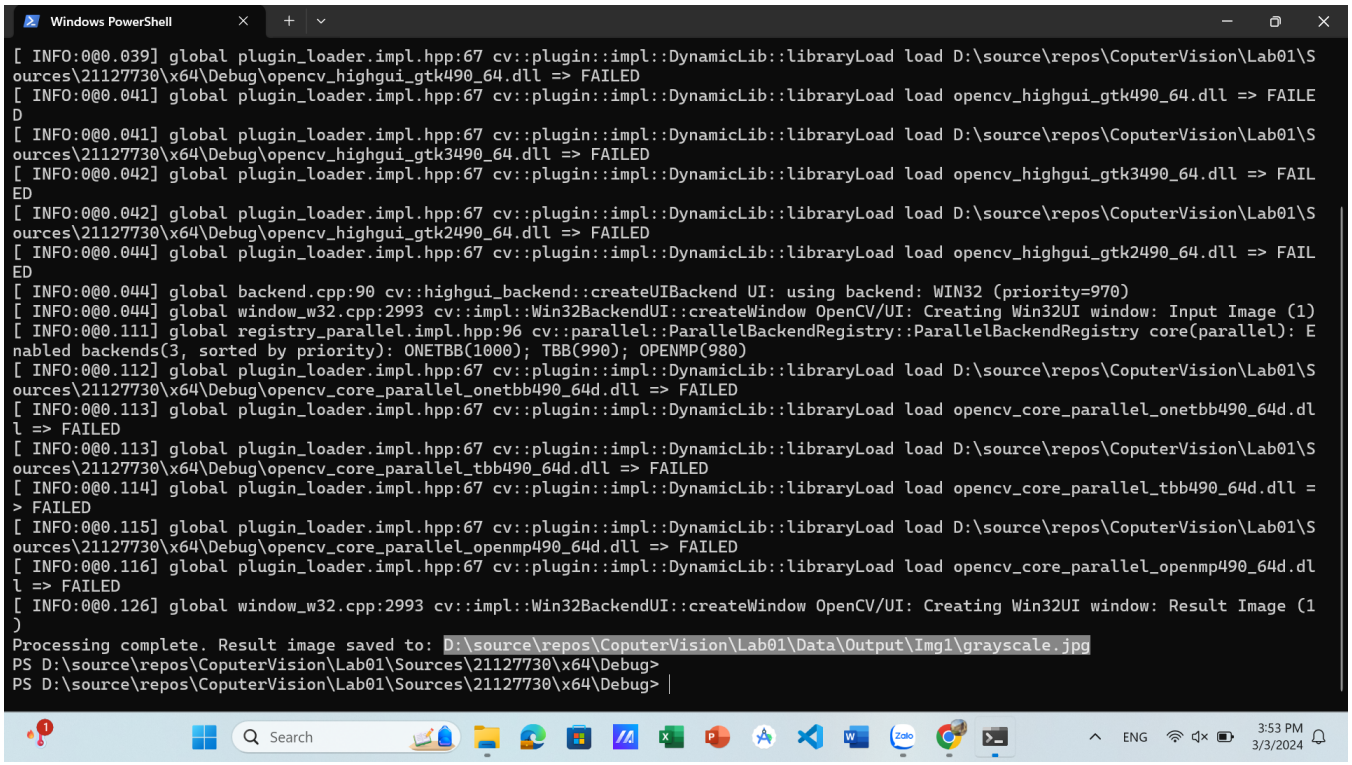

Figure 5: Output image saved successfully to the path