CS412 – 22125006 - Lab 1

CS412 – Individual Lab 1: Image Manipulation using OpenCV

Student Name: Ngo Hoang Bach

Date: October 23, 2020

1. Introduction

This report presents the implementation of a command-line image manipulation program using OpenCV in C++. The program provides five different image processing operations as specified in the lab requirements.

2. Program Overview

The program accepts command-line arguments and performs the following operations:

1. RGB to Grayscale conversion
2. Brightness adjustment with interactive trackbar
3. Contrast adjustment with interactive trackbar
4. Average filtering with kernel size control
5. Gaussian filtering with kernel size control

3. Implementation Details

3.1 Development Environment

- **Operating System**: macOS Sequoia
- **Programming Language**: C++11
- **Library**: OpenCV 4.12.0

- **Build System**: CMake 3.10+

3.2 Code Structure

- `main.cpp`: Main program logic with command-line parsing and image processing functions
- `CMakeLists.txt`: Build configuration file
- Trackbar callback functions for interactive parameter adjustment

3.3 Key Functions Implemented

RGB to Grayscale (`rgb2gray`)

```
cvtColor(originalImage, processedImage, COLOR_BGR2GRAY);
```

Converts BGR color image to grayscale using OpenCV's built-in conversion.

Brightness Adjustment (`brightness`)

```
processedImage = originalImage + Scalar(value, value, value);
```

Adds a constant value to all color channels for brightness modification.

Contrast Adjustment (`contrast`)

```
processedImage.at<Vec3b>(y, x)[c] = saturate_cast<uchar>(alpha * originall
mage.at<Vec3b>(y, x)[c]);
```

Multiplies pixel values by a scaling factor to adjust contrast.

Average Filter (`avg`)

```
blur(originalImage, processedImage, Size(value, value));
```

Applies box filter with adjustable kernel size.

Gaussian Filter (`gauss`)

```
GaussianBlur(originalImage, processedImage, Size(value, value), 0);
```

Applies Gaussian blur with adjustable kernel size.

4. Testing and Results

4.1 Test Images

- Input: `test_image.jpg`

- Output files generated for each operation

4.2 Functionality Verification

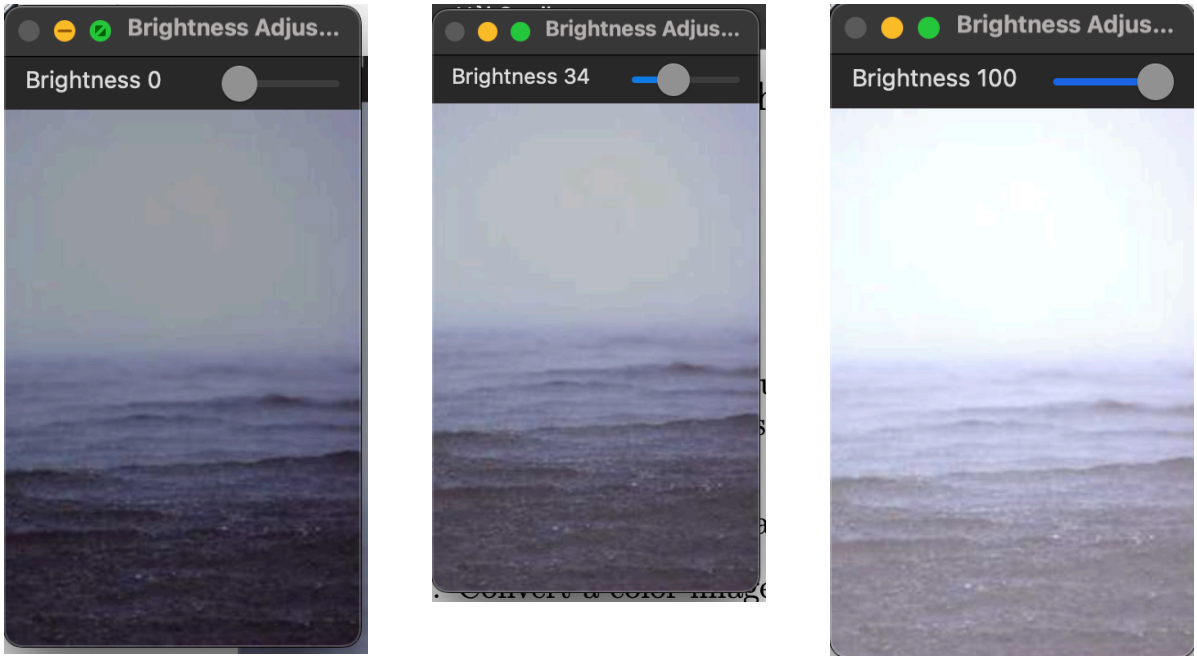
All five operations were successfully tested:

- ✅ RGB to grayscale conversion works correctly
- ✅ Brightness adjustment with trackbar (tested with default value +50)
- ✅ Contrast adjustment with trackbar (tested with default value 1.5x)
- ✅ Average filter with trackbar (tested with kernel size 5×5)
- ✅ Gaussian filter with trackbar (tested with kernel size 5×5)

4.3 Output Files Generated

- `output_rgb2gray.jpg`: Grayscale version

- `output_brightness.jpg`: Brightness adjusted



- `output_contrast.jpg`: Contrast adjusted

- `output_avg.jpg`: Average filtered

- `output_gauss.jpg`: Gaussian filtered

4.4 Sample Results

The processed images demonstrate the effectiveness of each operation:

- Grayscale conversion removes color information while preserving structure
- Brightness adjustment increases overall image intensity
- Contrast adjustment enhances the difference between light and dark areas

- Average filtering creates a smoothing effect by averaging neighboring pixels
- Gaussian filtering provides smoother blurring with edge preservation

5. Conclusion

The image manipulation program has been successfully implemented with all required features. The program demonstrates proper use of OpenCV functions for image processing and provides both command-line and interactive GUI interfaces. All operations work correctly and produce expected results as shown in the generated output files.