CSE 3113 / CSE 3214

INTRODUCTION TO DIGITAL IMAGE PROCESSING

SPRING 2024

Homework 1 Report

İkram Celal KESKİN– 200316059

Submission Date: 7 March 2024

Programming Environment

OS Name: Microsoft Windows 10 Home Single Language

OS Version: 10.0.19045 N/A Build 19045

System Model: VivoBook_ASUSLaptop X515DAP_D515DA

System Type: x64-based PC

Processor: AMD64 Family 23 Model 24 Stepping 1 AuthenticAMD ~2600 Mhz

Python Version: 3.11.0

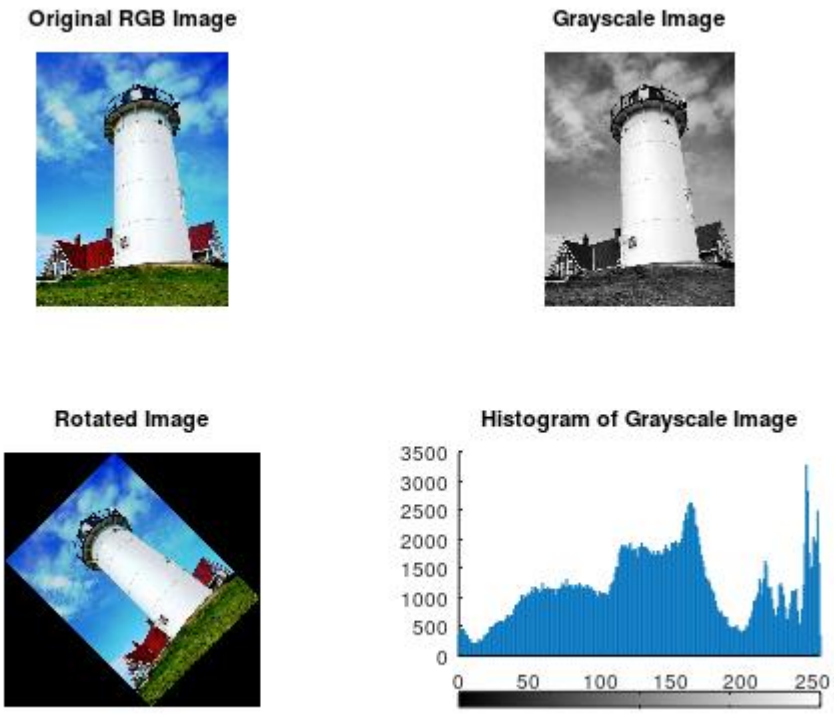
Pillow Version: 9.4.0

Matplotlib Version: 3.6.3

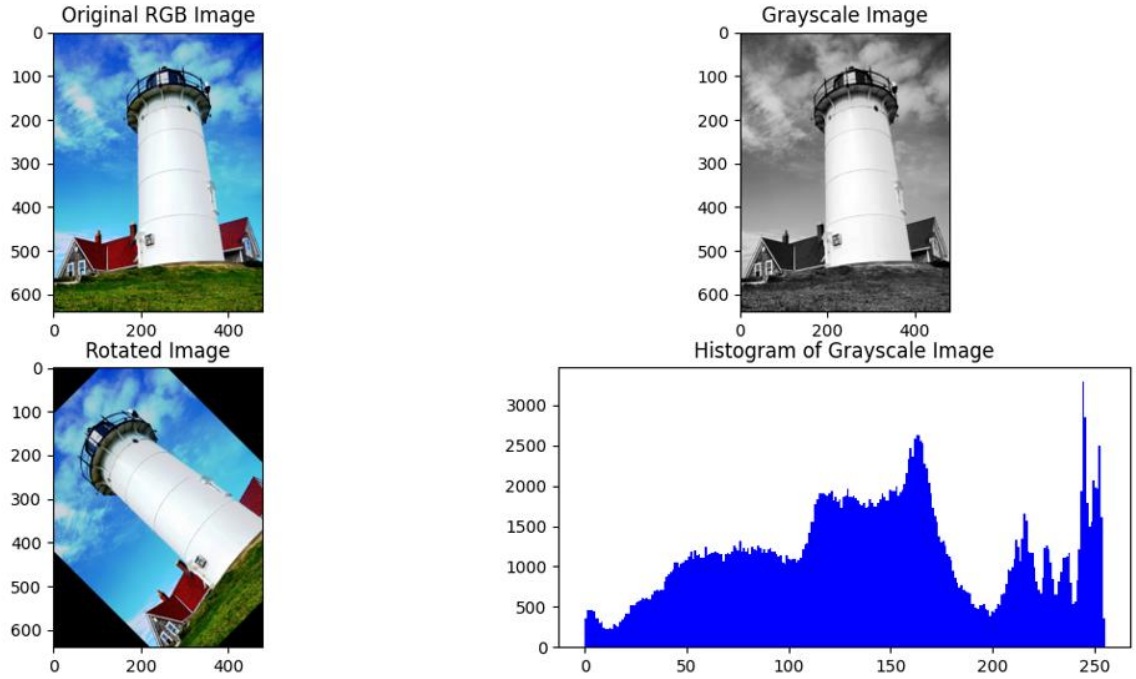
Numpy Version: 1.24.1

Output

This is the task 3 output (Octave)



This is the task 4 output (Python Pillow)



Reflections

I used gnu octave because matlab was not available. gnu octave codes are very similar to matlab. it took time to download the pillow library in python because my internet was not working.

Source Code

```
% Task 3 - Basic Image Processing in Octave


% Load the original RGB image
originalImage = imread('./images/lighthouse.png');


% Display the original RGB image
subplot(2, 2, 1);
```

```
imshow(originalImage);
title('Original RGB Image');

    % Convert the original RGB image to grayscale
grayImage = rgb2gray(originalImage);

    % Display the grayscale image
subplot(2, 2, 2);
imshow(grayImage);
title('Grayscale Image');

    % Load the image package
pkg load image

    % Rotate the original RGB image
rotatedImage = imrotate(originalImage, 45);

    % Display the rotated image
subplot(2, 2, 3);
imshow(rotatedImage);
title('Rotated Image');

    % Display the histogram of the grayscale image
subplot(2, 2, 4);
imhist(grayImage);
title('Histogram of Grayscale Image');
```

```
# Task 4 - Basic Image Processing with Python Pillow
from PIL import Image
import matplotlib.pyplot as plt
import numpy as np

# Load the original RGB image
original_image = Image.open('./images/lighthouse.png')

# Display the original RGB image
plt.subplot(2, 2, 1)
plt.imshow(np.array(original_image))
plt.title('Original RGB Image')

# Convert the original RGB image to grayscale
gray_image = original_image.convert('L')

# Display the grayscale image
plt.subplot(2, 2, 2)
plt.imshow(np.array(gray_image), cmap='gray')
plt.title('Grayscale Image')

# Rotate the original RGB image
rotated_image = original_image.rotate(45)

# Display the rotated image
plt.subplot(2, 2, 3)
plt.imshow(np.array(rotated_image))
```

```
plt.title('Rotated Image')


        # Display the histogram of the grayscale image
plt.subplot(2, 2, 4)
plt.hist(np.array(gray_image).ravel(), bins=256, color='blue')
plt.title('Histogram of Grayscale Image')

plt.show()
```