**Elective 3**

Laboratory Activity No. 1

**Image Acquisition and Manipulation**

|  |
| --- |
|  |

Score

*Submitted by:*

**Leonardo, Aibel RenzoT.**

**S 7:00PM – 4:00PM / CPE 0332.1-1**

*Date Submitted*

**24-07-2024**

*Submitted to:*

**Engr. Maria Rizette H. Sayo**

1. Objectives

This laboratory activity aims to implement the principles and techniques of image acquisition through MATLAB/Octave and open CV using Python

* Acquire the image.
* Rotate the image by 30 degrees.
* Flip the image horizontally.

1. Methods
2. Perform a task given in the presentation

* Copy and paste your MATLAB code
  + img = imread('flower.jpg');
  + % Rotate by 45 degrees
  + rotImg = imrotate(img, 30);
  + % Flip horizontally
  + flipImg = fliplr(rotImg);
  + % Display results
  + figure(1);
  + plot(1, 1);
  + imshow(img);
  + title('Original Image');
  + figure(2);
  + plot(1, 1);
  + imshow(rotImg);
  + title('Rotated 45°');
  + figure(3);
  + plot(1, 1);
  + imshow(flipImg);
  + title('Rotated & Flipped');

Supplementary Activity

* Write a Python program that will implement the output in Method A.
* import cv2
* import numpy as np
* img = cv2.imread('flower.jpg')
* (h, w) = img.shape[:2]
* center = (w / 2, h / 2)
* M = cv2.getRotationMatrix2D(center, 30, 1.0)
* rotated\_img = cv2.warpAffine(img, M, (w, h))
* flipped\_img = cv2.flip(rotated\_img, 1)
* max\_height = max(img.shape[0], rotated\_img.shape[0], flipped\_img.shape[0])
* combined\_width = img.shape[1] + rotated\_img.shape[1] + flipped\_img.shape[1]
* combined\_img = np.zeros((max\_height, combined\_width, 3), dtype = np.uint8)
* combined\_img[:img.shape[0], :img.shape[1]] = img
* combined\_img[:rotated\_img.shape[0], img.shape[1]:img.shape[1] + rotated\_img.shape[1]] = rotated\_img
* combined\_img[:flipped\_img.shape[0], img.shape[1] + rotated\_img.shape[1]:] = flipped\_img
* cv2.imshow('Combined Image', combined\_img)
* cv2.imwrite('combined\_image.jpg', combined\_img)
* cv2.waitKey(0)
* cv2.destroyAllWindows()

1. Results
2. Copy/crop and paste your results. Label each output (Figure1, Figure2, Figure3)

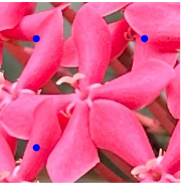
picture file: flower.jpg

Figure 1: Acquire an Image of a Flower



Figure 2: Rotate by 30 degrees

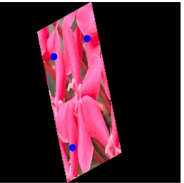


Figure 3: Flip horizontally

**MATLAB:**

A screenshot of a computer

Description automatically generated

**Figure 1**

A screenshot of a computer

Description automatically generated

**Figure 2**

A screenshot of a computer

Description automatically generated

**Figure 3**

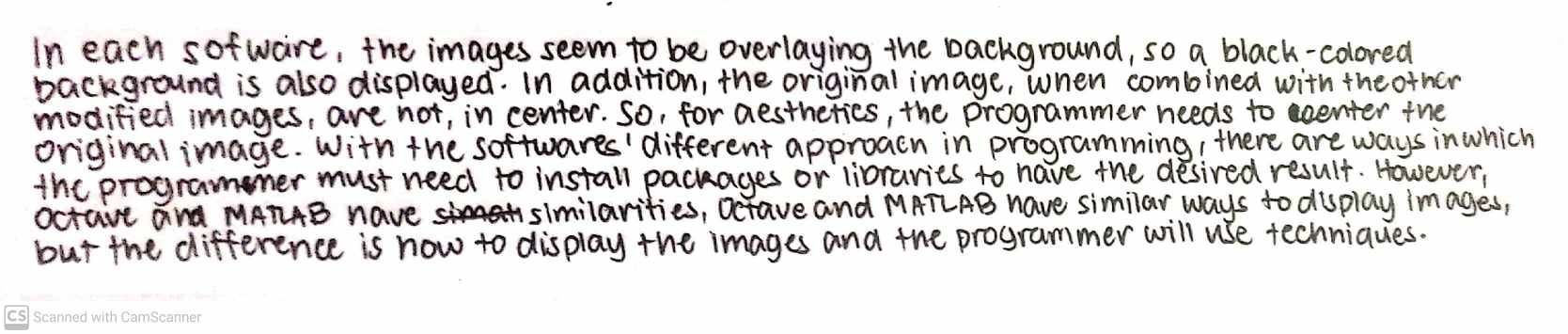
**Python:**

A close up of a flower

Description automatically generated

**Figure 1, 2 and 3**

1. Visualize the results, analyze and interpret:



IV. Conclusion

In conclusion, this laboratory exercise effectively illustrated the basic methods of image display and manipulation using MATLAB/Octave and Python. Successfully completing these tasks of rotation and flipping an image. It is revealed that while MATLAB and Octave share similarities in image processing, they differ in their approach to displaying and centering images.

In Python’s method, using the Open CV library, provided an alternative approach to these tasks, offering insights into how different tools handle image manipulation [1]. The results showed the importance of understanding each software’s specific requirements and the role of additional libraries or packages in achieving desired outcomes. This exercise highlighted the flexibility of image processing tools and the need for careful programming to produce clear and visually appealing results

**References**

[1] *OpenCV Python Tutorial. (2020, January 30). GeeksforGeeks. https://www.geeksforgeeks.org/opencv-python-tutorial/*