

INT3404E 20 - Image Processing: Homework 1

Pham Duc Trung - 21021548

11th April 2024

1 The original image

This is the original image to be processed (The color is not correct due to running on Google Colaboratory(?)).

Original Image



Figure 1: The original image

2 Function *grayscale_image*

This function will convert an image to grayscale. Convert the original image to a grayscale image. In a grayscale image, the pixel value of the 3 channels will be the same for a particular X, Y coordinate. The argument of this function is the *image* to be grayscaled.

Code:

```
import cv2
import matplotlib.pyplot as plt
import numpy as np

5 def grayscale_image(image):
    row, column, channel = image.shape
    img_gray = np.zeros((row, column, channel), dtype=np.uint8)
    for i in range(row):
        for j in range(column):
10         R, G, B = image[i, j]
            gray_value = 0.299 * R + 0.587 * G + 0.114 * B
            img_gray[i, j] = gray_value
    return img_gray
```



Figure 2: Result of the function *grayscale_image*

3 Function *flip_image*

This function will flip an image horizontally using OpenCV. The argument of this function is the *image* to be flipped.

Code:

```
import cv2
import matplotlib.pyplot as plt
import numpy as np

5 def flip_image(image):
    flipped_image = cv2.flip(image, 1)
    return flipped_image
```



Figure 3: Result of the function *flip_image*

4 Function *rotate_image*

This function will rotate an image using OpenCV. The arguments of this function are the *image* to be rotated and the *angle* of the rotation which is in degrees.

Code:

```
import cv2
import matplotlib.pyplot as plt
import numpy as np

5 def rotate_image(image, angle):
    height, width = image.shape[:2]
    rotation_matrix = cv2.getRotationMatrix2D((width / 2, height / 2), angle, 1)
    rotated_image = cv2.warpAffine(image, rotation_matrix, (width, height))
    return rotated_image
```



Figure 4: Result of the function *rotate_image* with angle 45°