INT3404E 20 - Image Processing: Homeworks week 1

Duong Bao Long - 21021514

1 Function flip_image:

Flip image code

```python
def flip_image(image):
    """
    Flip an image horizontally using OpenCV
    """
    result = cv2.flip(image, 1)
    result_rgb = cv2.cvtColor(result, cv2.COLOR_BGR2RGB)
    return result_rgb
```

Figure 1: Flip image result (horizontal)

2 Function rotate_image:

Rotate image code

```python
def rotate_image(image, angle):
    """
    Rotate an image using OpenCV. The angle is in degrees
    """

    image_center = tuple(np.array(image.shape[1::-1]) / 2)
    rot_mat = cv2.getRotationMatrix2D(image_center, angle, 1.0)
    result = cv2.warpAffine(image, rot_mat, image.shape[1::-1], flags=cv2.INTER_LINEAR)
    result_rgb = cv2.cvtColor(result, cv2.COLOR_BGR2RGB)
    return result_rgb
```

Figure 2: Rotate 45° image result

3 Function grayscale_image:

Grayscale image code

```python
def grayscale_image(image):
    """
    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 equation for the pixel value
    [1] is given by:
        p = 0.299R + 0.587G + 0.114B
    Where the R, G, B are the values for each of the corresponding channels.
    We will do this by creating an array called img_gray with the same shape as img
    """
    pixel_map = image
    width, height, cl = image.shape
    for i in range(width):
        for j in range(height):
            # getting the RGB pixel value.
            r, g, b = image[i][j]
            # Apply formula of grayscale:
            grayscale = 0.299 * r + 0.587 * g + 0.114 * b
            # setting the pixel value.
            pixel_map[i, j] = [int(grayscale), int(grayscale), int(grayscale)]
    return pixel_map
```

Figure 3: Grayscale image result