

INT3404E 20 - Image Processing: Homeworks 1

Ngô Lê Hoàng - 22028042

1 Đề bài

Sử dụng thư viện Numpy, OpenCV, Matplotlib để thực hiện các thao tác với ảnh. Các thao tác này bao gồm load ảnh, biểu diễn ảnh, chuyển ảnh thành ảnh xám, lưu ảnh, lật ảnh, xoay ảnh.

Báo cáo kết quả của các hàm đã thực hiện.

2 Kết quả

2.1 Load image

Ảnh đầu vào sử dụng color model là BGR nên cần phải convert sang RGB model

```
def load_image(image_path):  
    return cv2.cvtColor(cv2.imread(image_path), cv2.COLOR_BGR2RGB)
```

2.2 Display image

```
def display_image(image, title="Image"):  
    plt.imshow(image)  
    plt.title(title)  
    plt.show()
```

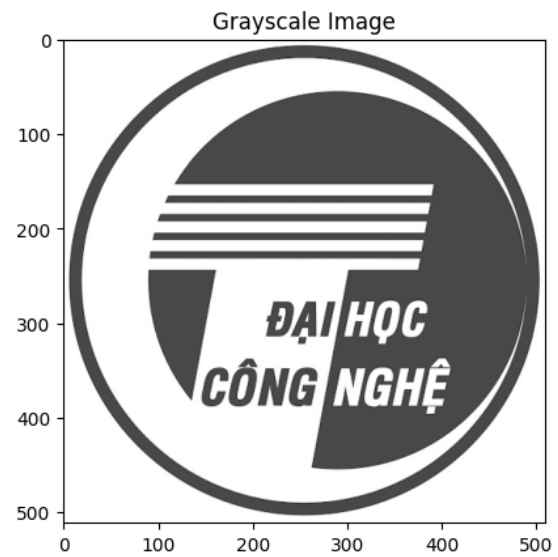
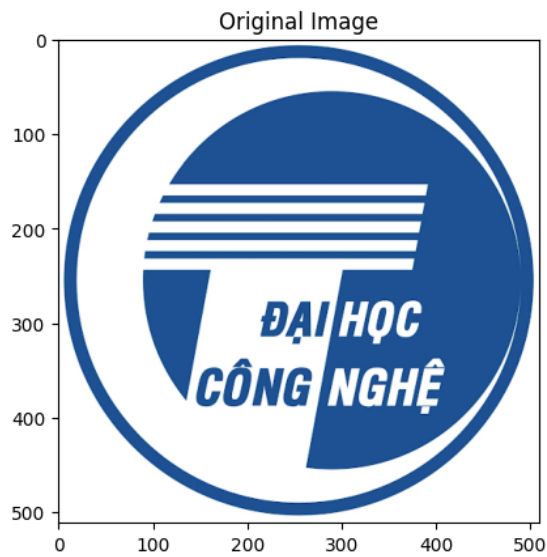
2.3 Save image

```
def save_image(image, output_path):  
    cv2.imwrite(output_path, image)
```

2.4 Grayscale image

Mỗi giá trị pixel được tính theo công thức $p = 0.299R + 0.587G + 0.114B$ với R,G,B là các giá trị tương ứng với 3 kênh màu.

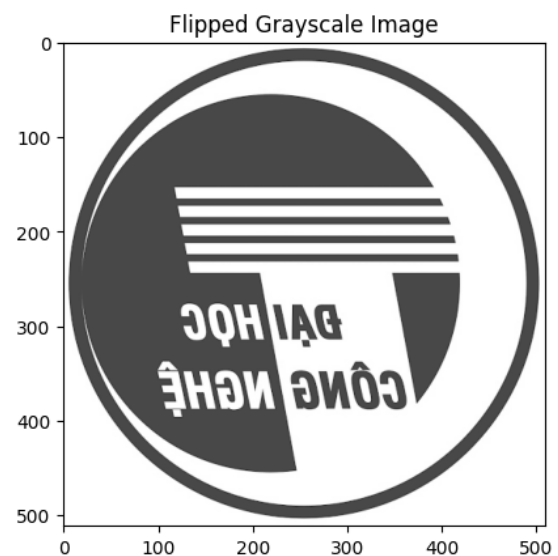
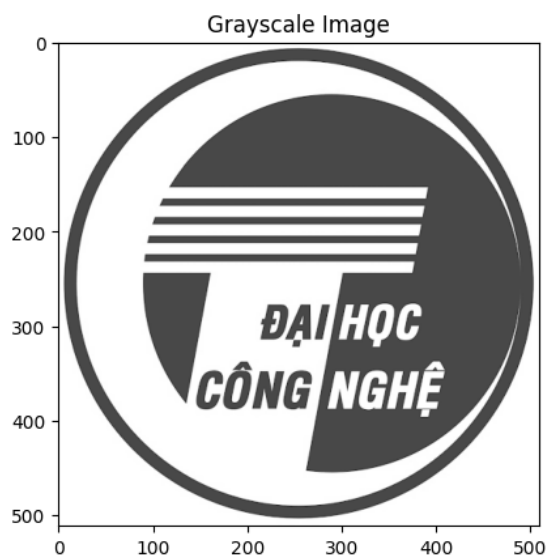
```
def grayscale_image(image):  
    shape = image.shape  
    h, w = shape[0], shape[1]  
    img_gray = image  
    for i in range(h):  
        for j in range(w):  
            r,g,b = image[i][j]  
            img_gray[i][j] = 0.299 * r + 0.587 * g + 0.114 * b  
    return img_gray
```



2.5 Flip image

Để lật ảnh sử dụng hàm `cv2.flip`, tham số axis của hàm flip để bằng 1 ứng với lật theo chiều ngang.

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



2.6 Rotate image

Trước tiên tính tâm xoay rồi tạo một ma trận xoay (rotation matrix) bằng

hàm *cv2.getRotationMatrix2D*. Sau đó thực hiện phép xoay với hàm *cv2.warpAffine*.

```
def rotate_image(image, angle):  
    shapes = image.shape  
    h, w = shapes[0], shapes[1]  
    center = (h / 2, w / 2)  
    rotate_matrix = cv2.getRotationMatrix2D(center, angle, 1)  
    rotated_image = cv2.warpAffine(image, rotate_matrix, (h,w))  
    return rotated_image
```

