พสิษฐ์ โวศรี 6434453423

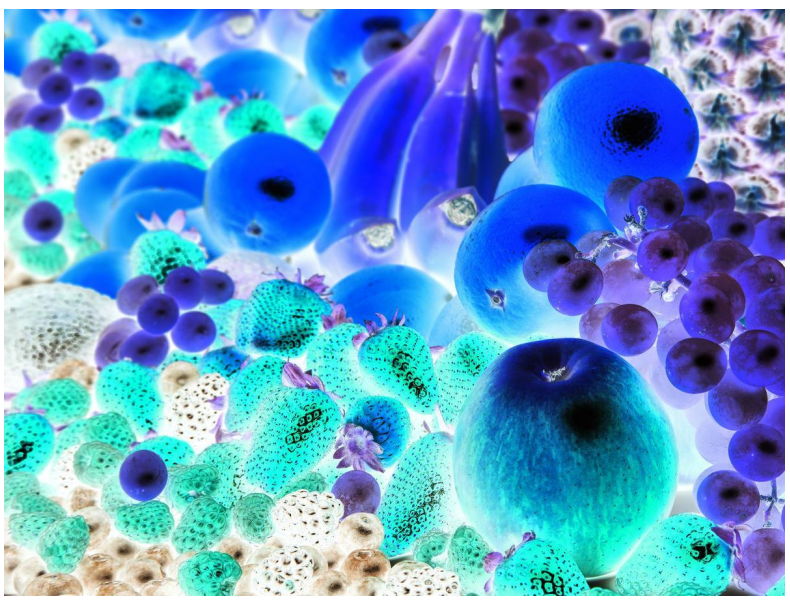
1. Convert image to its complementary colors

```python
import numpy as np
import matplotlib.pyplot as plt
from skimage.io import imread, imsave
import skimage

def invert_image(img_path):
    image = imread(img_path)
    image = skimage.util.invert(image)
    file_name = img_path.strip(".jpg").split("/")[-1]
    imsave(f"ImageProcessing/assign5/output/{file_name}_inverted.jpg",image)
    plt.imshow(image)
    plt.show()

invert_image("ImageProcessing/assign5/fruit.jpg")
```

OUTPUT

Original

Transformed



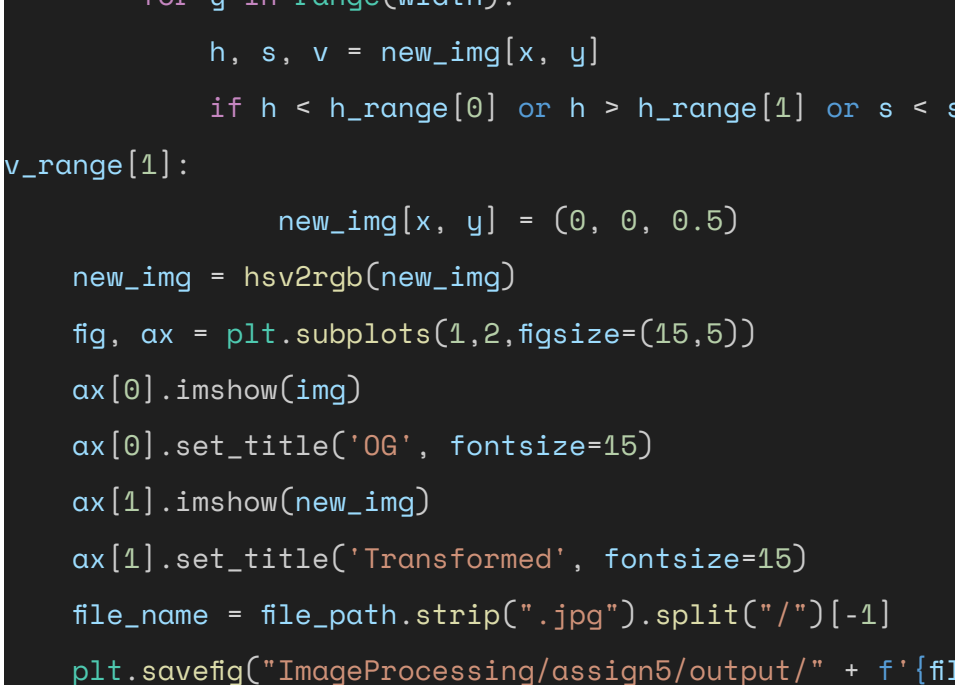
2. Color Slicing (RGB cube vs HSV)

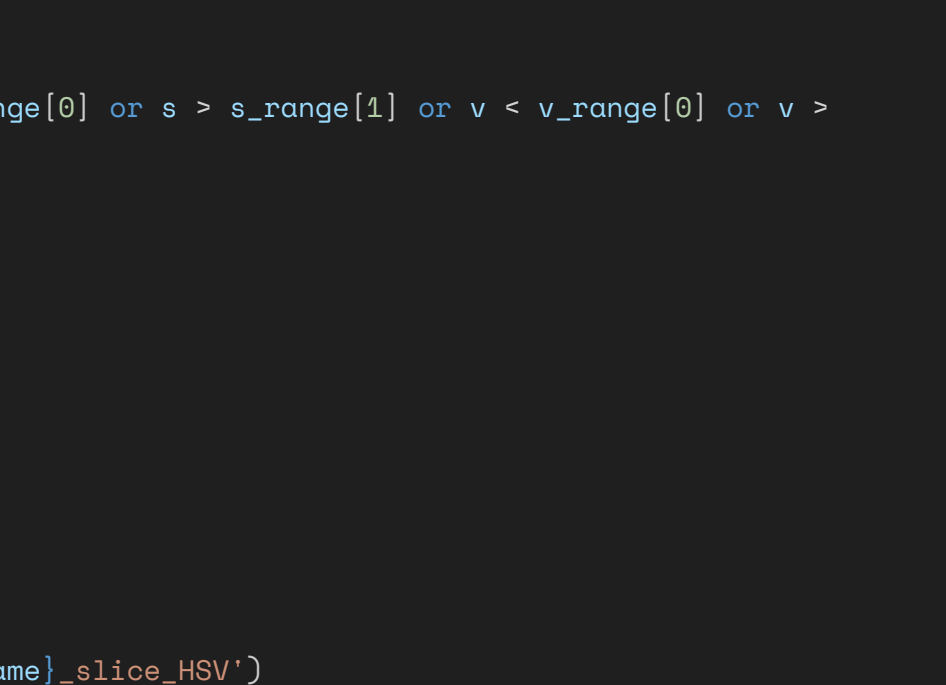
```python
import matplotlib.pyplot as plt
from skimage.io import imread
from skimage.color import rgb2hsv, hsv2rgb
import skimage

def slice_color_cube(file_path: str, color: tuple[list], w: int):
    img = imread(file_path)
    new_img = skimage.util.img_as_float32(img) # normalize
    height, width, _ = img.shape
    for x in range(height):
        for y in range(width):
            r = new_img[x, y]
            # check if r in cube
            in_cube = True
            for j in range(3):
                if abs(r[j]-color[j]) > (w/2):
                    in_cube = False
                    break
            if not in_cube:
                new_img[x, y] = (0.5, 0.5, 0.5)
    # convert to 8-bit image
```

```
    new_img = skimage.util.img_as_ubyte(new_img)
    fig, ax = plt.subplots(1,2,figsize=(15,5))
    ax[0].imshow(img)
    ax[0].set_title('OG', fontsize=15)
    ax[1].imshow(new_img)
    ax[1].set_title('Transformed', fontsize=15)
    file_name = file_path.strip(".jpg").split("/")[-1]
    plt.savefig("ImageProcessing/assign5/output/" + f'{file_name}_slice_cube')
    plt.show()

# slice_color_cube("ImageProcessing/assign5/oranges.jpg", (0.9411754706, 0.5490196078, 0) , 0.588235291)

def slice_color_HSV(file_path: str, h_range: tuple[int], s_range: tuple[int], v_range: tuple[int]):
    img = imread(file_path)
    new_img = rgb2hsv(img)
    height, width, _ = img.shape
    for x in range(height):
        for y in range(width):
            h, s, v = new_img[x, y]
            if h < h_range[0] or h > h_range[1] or s < s_range[0] or s > s_range[1] or v < v_range[0] or v >
v_range[1]:
                new_img[x, y] = (0, 0, 0.5)
    new_img = hsv2rgb(new_img)
    fig, ax = plt.subplots(1,2,figsize=(15,5))
    ax[0].imshow(img)
    ax[0].set_title('OG', fontsize=15)
    ax[1].imshow(new_img)
    ax[1].set_title('Transformed', fontsize=15)
    file_name = file_path.strip(".jpg").split("/")[-1]
    plt.savefig("ImageProcessing/assign5/output/" + f'{file_name}_slice_HSV')
    plt.show()

# slice_color_HSV("ImageProcessing/assign5/oranges.jpg", (0, 0.138), (0.3, 1), (0.30, 1))
```

OUTPUT

RGB cube : rgb = (240, 140, 0), cube width = 150





HSV color space : Hue range 0° to 50°, Saturation 30% to 100%, Value 30% to 100%

พสิษฐ์ โวศรี 6434453423