

Name: Sumanasekara W.K.G.G.

Index: 190610E

```
In [ ]: import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
```

```
In [ ]: spider = cv.imread(r"spider.png", cv.IMREAD_GRAYSCALE)
assert spider is not None

fig1, ax1 = plt.subplots(1, 2, figsize = (17, 10))
fig2, ax2 = plt.subplots(1, 2, figsize = (17, 10))

axs = (ax1, ax2)
gamma = (0.2, 0.8, 1.2, 2)
for index, gamma in enumerate(gamma):
    look_up = np.array([((i/255)**gamma)*255 for i in range(256)]).astype(np.uint8)
    transformed_image = cv.LUT(spider, look_up)

    axs[index//2][index%2].imshow(transformed_image, cmap="gray", vmin=0, vmax=255)
    axs[index//2][index%2].get_xaxis().set_visible(False)
    axs[index//2][index%2].get_yaxis().set_visible(False)
    axs[index//2][index%2].set_title("Gamma corrected image (gamma = {})".format(gamma))

plt.show()
```

Gamma corrected image (gamma = 0.2)



Gamma corrected image (gamma = 0.8)



Gamma corrected image (gamma = 1.2)



Gamma corrected image (gamma = 2)



```
In [ ]: t1 = np.linspace(0, 100, 51)
t2 = np.linspace(100, 255, 150)
t3 = np.linspace(255, 255, 55)

fig, ax = plt.subplots(1, 2, figsize = (20, 6))
```

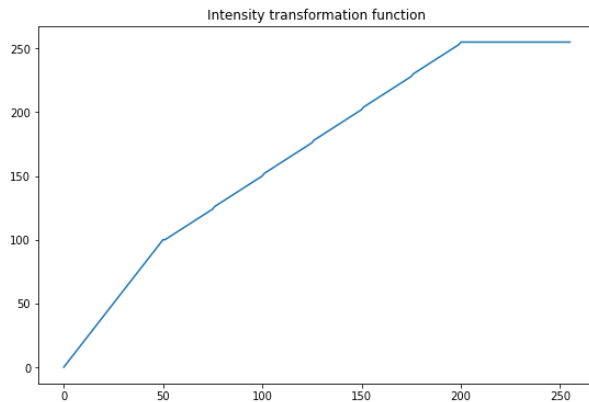
```

transform = np.concatenate((t1, t2, t3), axis=0).astype(np.uint8)
ax[0].plot(transform)
ax[0].set_title("Intensity transformation function")

transformed_image = cv.LUT(spider, transform)
ax[1].imshow(transformed_image, cmap="gray", vmin=0, vmax=255)
ax[1].get_xaxis().set_visible(False)
ax[1].get_yaxis().set_visible(False)
ax[1].set_title("Transformed image")

plt.show()

```



In []:

```

shells = cv.imread(r"shells.tif", cv.IMREAD_GRAYSCALE)
assert shells is not None

original_hist = cv.calcHist([shells], [0], None, [256], [0, 256])
equalized_img = cv.equalizeHist(shells)
equalized_hist = cv.calcHist([equalized_img], [0], None, [256], [0, 256])

fig1, ax1 = plt.subplots(1, 2, figsize=(20, 5))

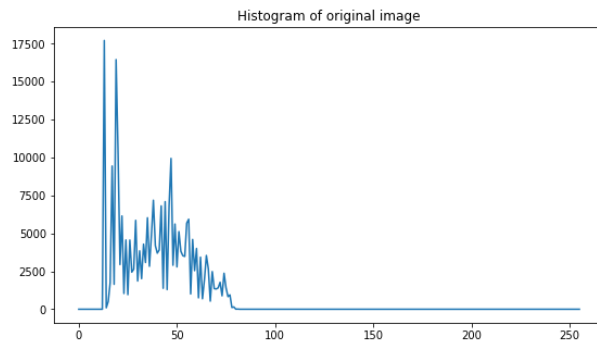
ax1[0].plot(original_hist)
ax1[0].set_title("Histogram of original image")
ax1[1].plot(equalized_hist)
ax1[1].set_title("Histogram of equalized image")

fig2, ax2 = plt.subplots(1, 2, figsize=(10, 5))
ax2[0].imshow(shells, cmap="gray", vmin=0, vmax=255)
ax2[0].get_xaxis().set_visible(False)
ax2[0].get_yaxis().set_visible(False)
ax2[0].set_title("Original image")

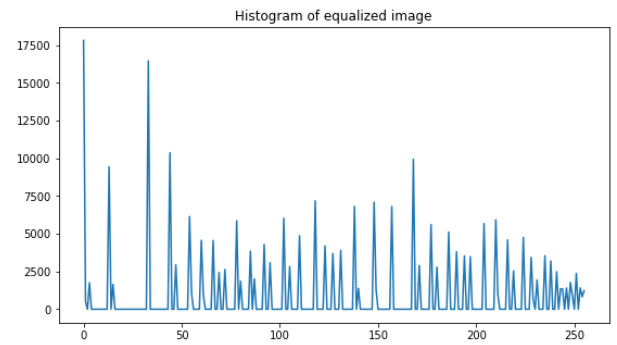
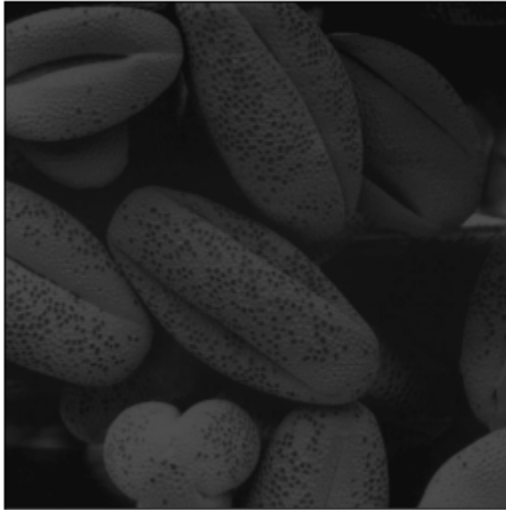
ax2[1].imshow(equalized_img, cmap="gray", vmin=0, vmax=255)
ax2[1].get_xaxis().set_visible(False)
ax2[1].get_yaxis().set_visible(False)
ax2[1].set_title("Equalized image")

plt.show()

```



Original image



Equalized image



In []:

```

zion_pass = cv.imread(r"zion_pass.jpg", cv.IMREAD_COLOR)
zion_pass_RGB = cv.cvtColor(zion_pass, cv.COLOR_BGR2RGB)

zion_pass_HSV = cv.cvtColor(zion_pass, cv.COLOR_BGR2HSV)
factor = 30
zion_pass_HSV[:, :, 1] = cv.add(zion_pass_HSV[:, :, 1], factor)
zion_pass_enhanced_RGB = cv.cvtColor(zion_pass_HSV, cv.COLOR_HSV2RGB)

fig, ax = plt.subplots(1, 2, figsize=(15, 10))

ax[0].imshow(zion_pass_RGB)
ax[0].get_xaxis().set_visible(False)
ax[0].get_yaxis().set_visible(False)
ax[0].set_title("Original image")

ax[1].imshow(zion_pass_enhanced_RGB)
ax[1].get_xaxis().set_visible(False)
ax[1].get_yaxis().set_visible(False)
ax[1].set_title("After enhancing the saturation by {}".format(factor))

plt.show()

```

Original image



After enhancing the saturation by 30



In []:

```

zion_pass = cv.imread(r"zion_pass.jpg", cv.IMREAD_COLOR)
zion_pass_RGB = cv.cvtColor(zion_pass, cv.COLOR_BGR2RGB)

zion_pass_HSV = cv.cvtColor(zion_pass, cv.COLOR_BGR2HSV)
factor = -15
zion_pass_HSV = cv.add(zion_pass_HSV, factor)
zion_pass_hue_changed_RGB = cv.cvtColor(zion_pass_HSV, cv.COLOR_HSV2RGB)

fig, ax = plt.subplots(1, 2, figsize=(15, 10))

ax[0].imshow(zion_pass_RGB)
ax[0].get_xaxis().set_visible(False)
ax[0].get_yaxis().set_visible(False)
ax[0].set_title("Original image")

ax[1].imshow(zion_pass_hue_changed_RGB)
ax[1].get_xaxis().set_visible(False)
ax[1].get_yaxis().set_visible(False)
ax[1].set_title("After changing the hue by {}".format(factor))

plt.show()

```

Original image



After changing the hue by -15

