

EN2550_Exercise2_190621M

February 9, 2022

0.0.1 Exercise-02

0.0.2 Index No - 190621M

0.0.3 Name - K. Thanushan

```
[ ]: #Question 1
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt

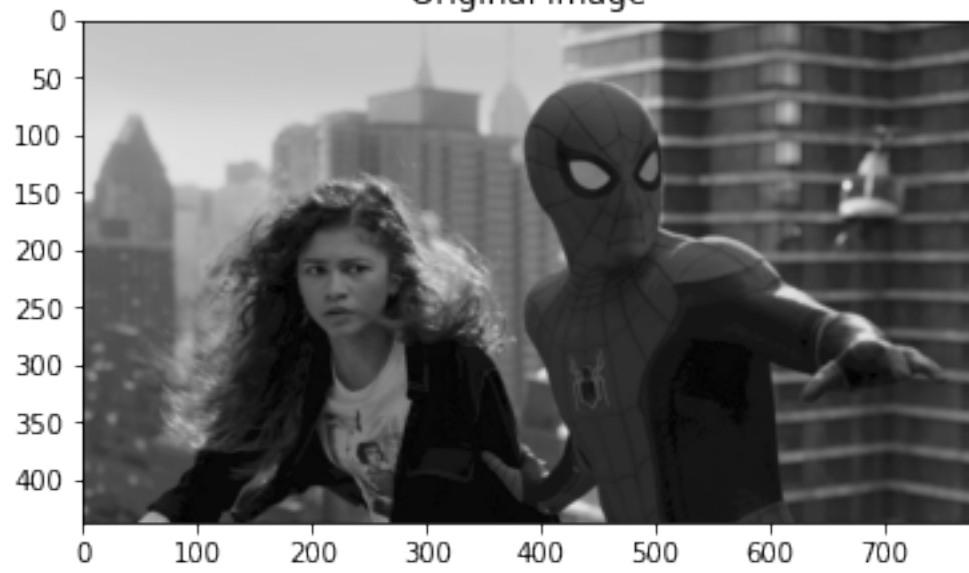
O_image = cv.imread(r'spider.png', cv.IMREAD_GRAYSCALE)
assert O_image is not None
img = cv.cvtColor(O_image, cv.COLOR_BGR2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Original Image')
plt.show()

gamma_List = [0.2, 0.8, 1.2, 2]
for gamma in gamma_List:
    t = np.array([(p/255)**gamma*255 for p in range(0,256)]).astype(np.uint8)
    g = cv.LUT(O_image,t)

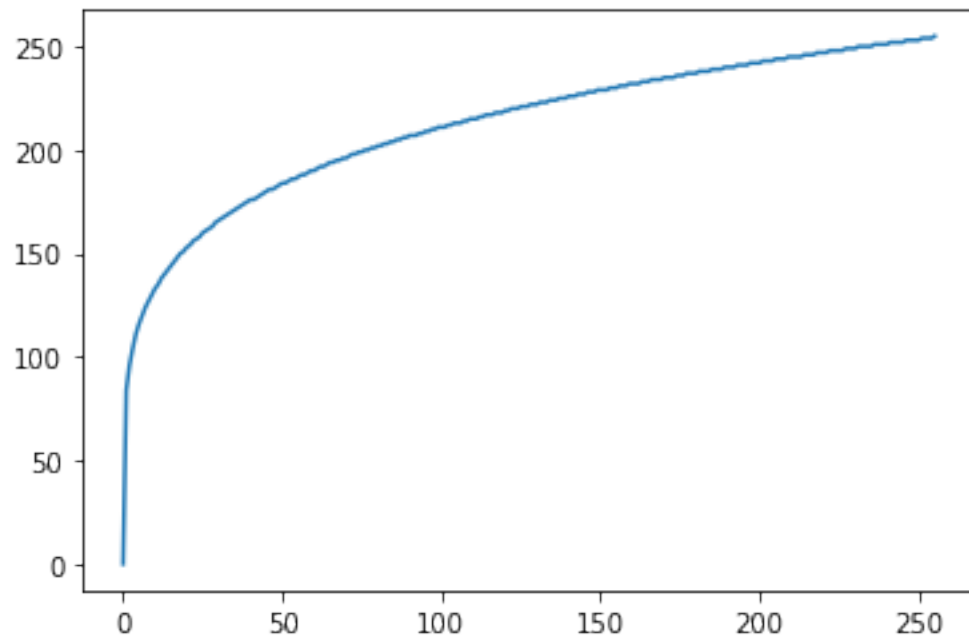
    fig, ax = plt.subplots()
    ax.plot(t)
    ax.set_title(gamma)

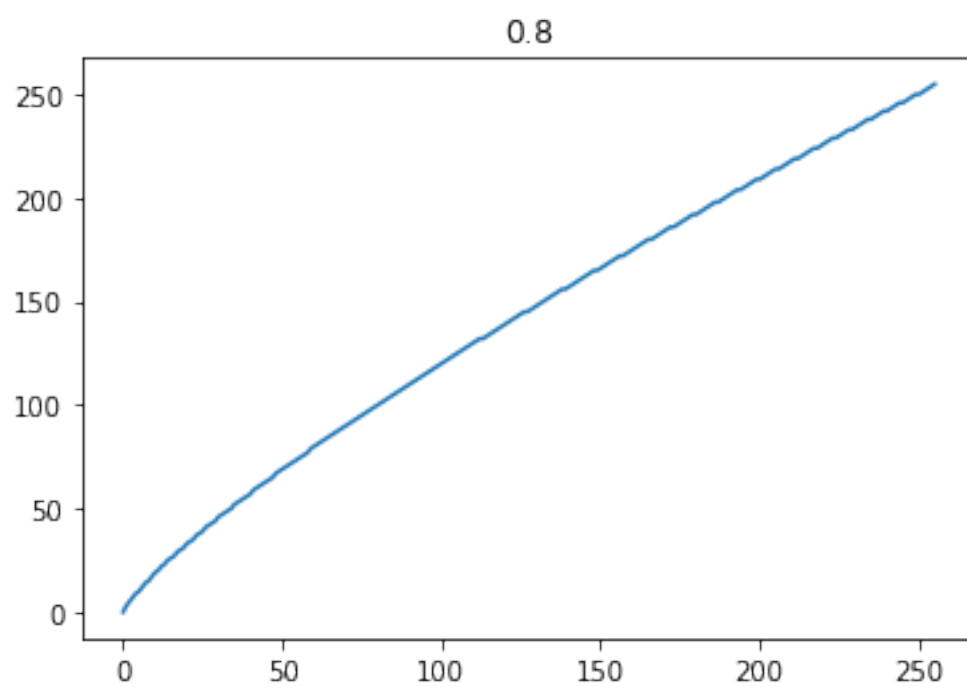
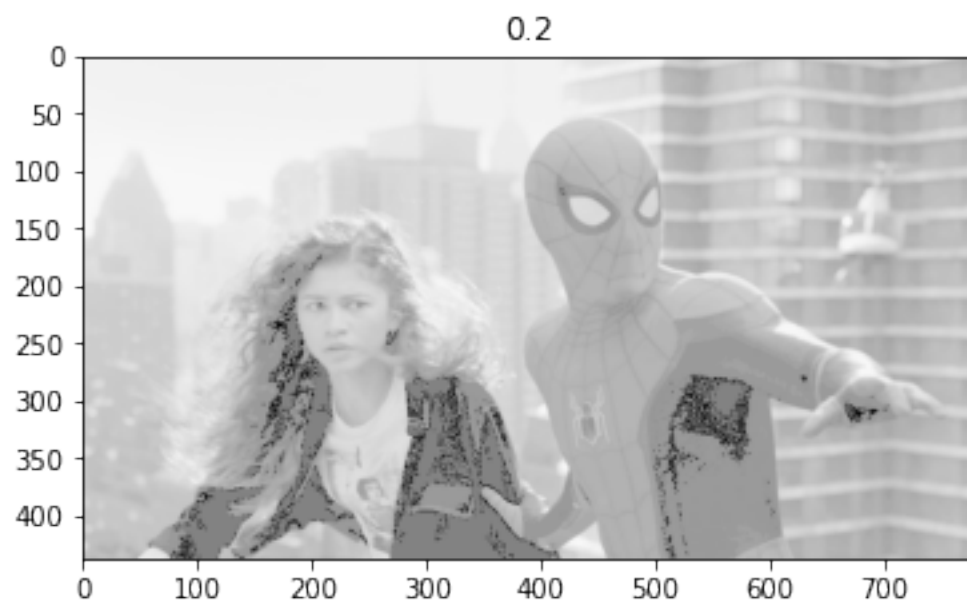
    img = cv.cvtColor(g, cv.COLOR_BGR2RGB)
    fig, ax = plt.subplots()
    ax.imshow(img)
    ax.set_title(gamma)
    plt.show()
```

Original Image

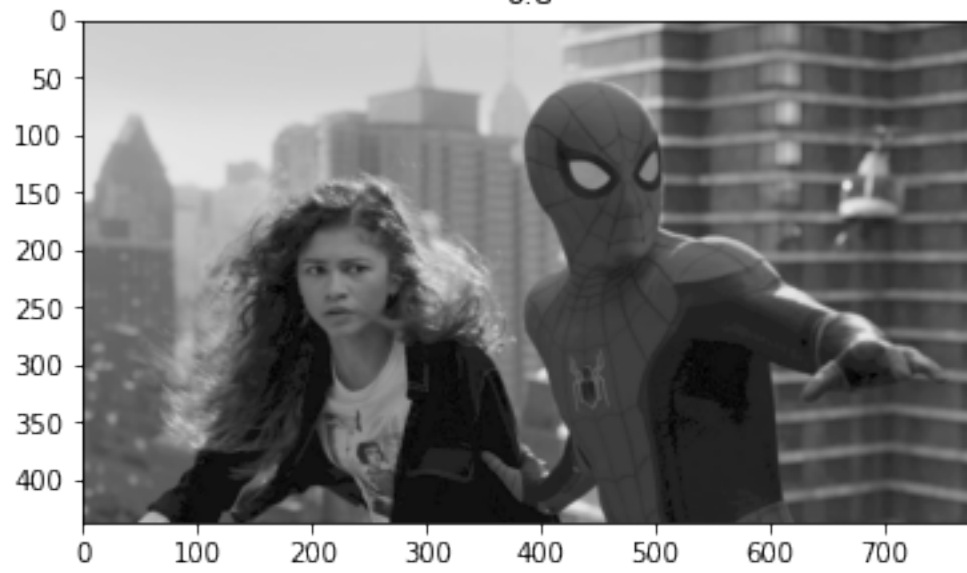


0.2

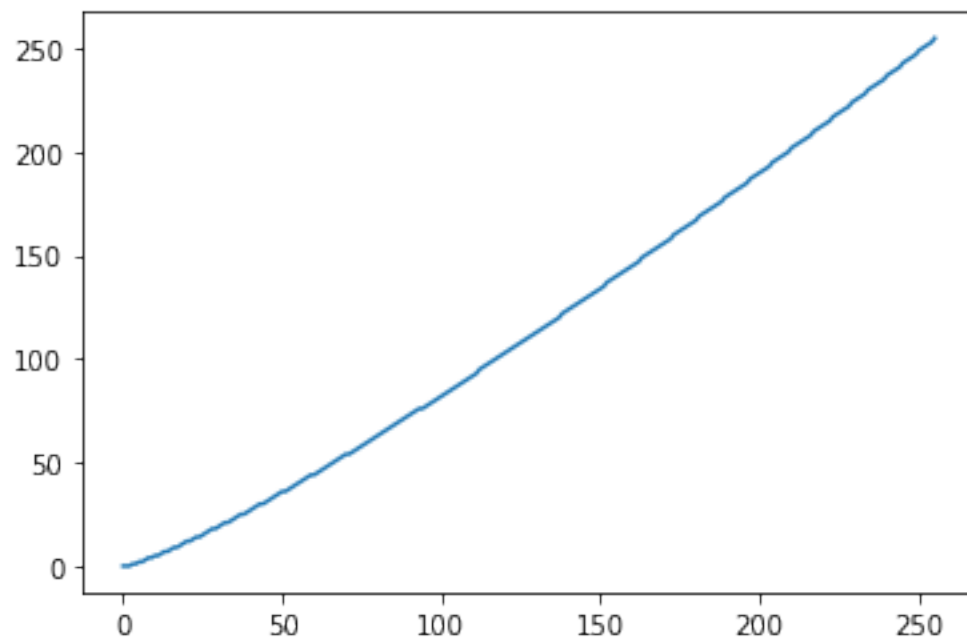




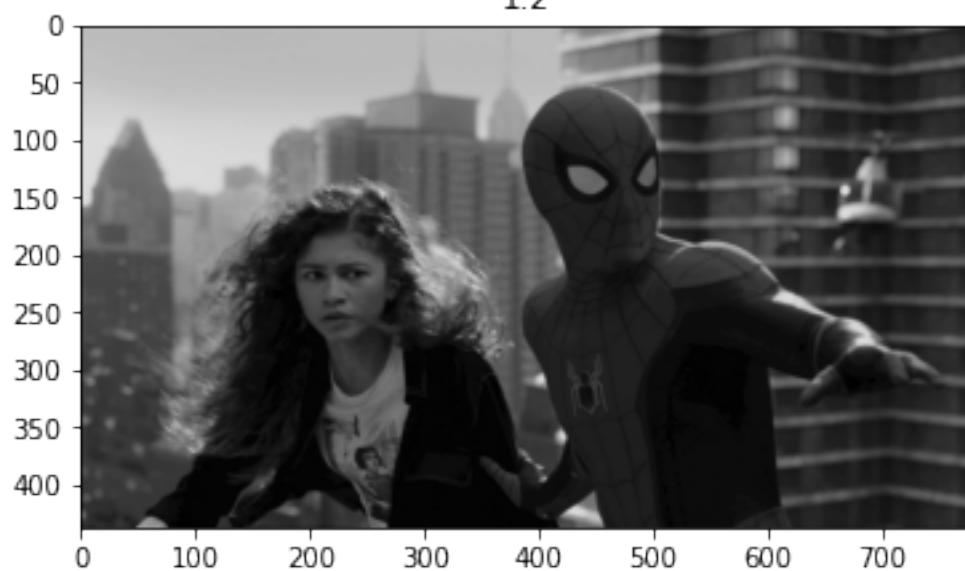
0.8



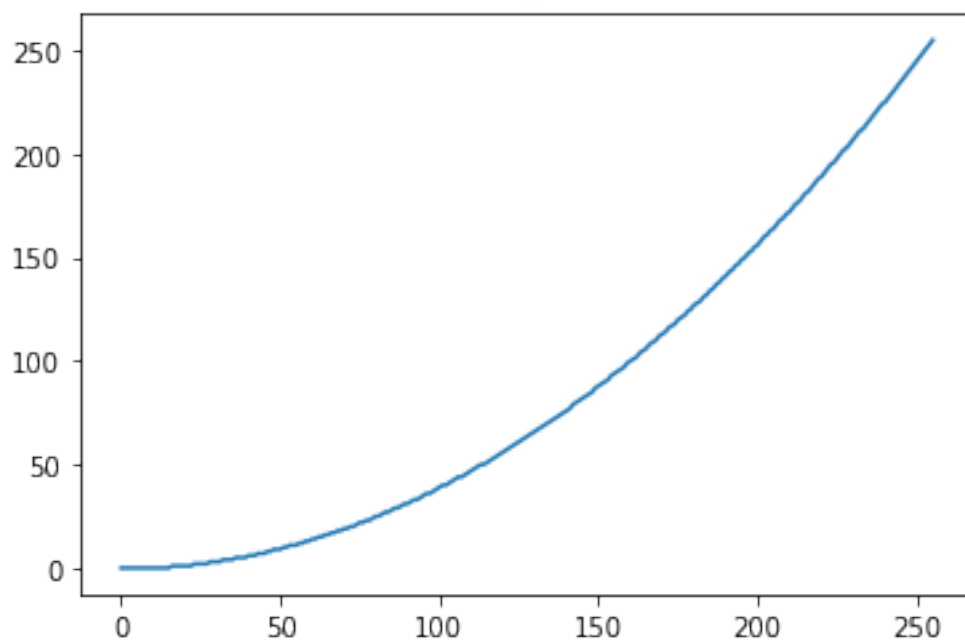
1.2

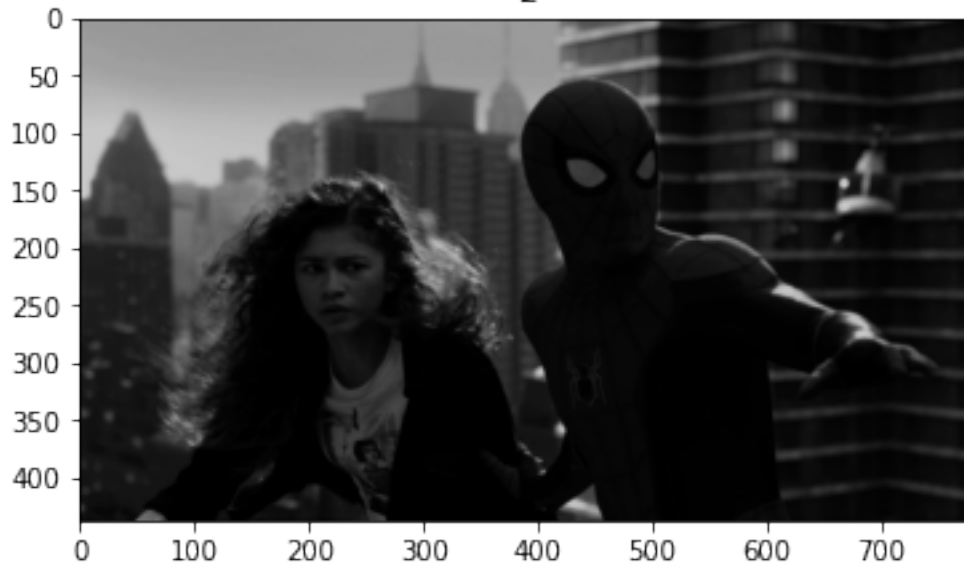


1.2



2





```
[ ]: #Question 2
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt

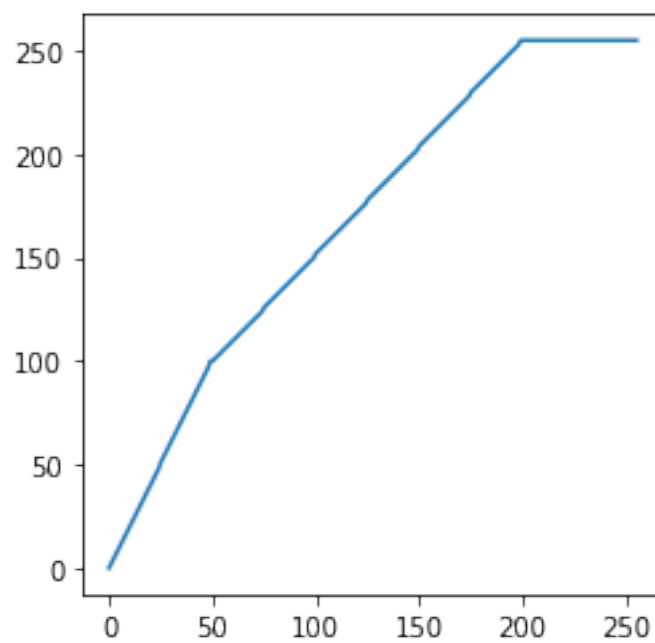
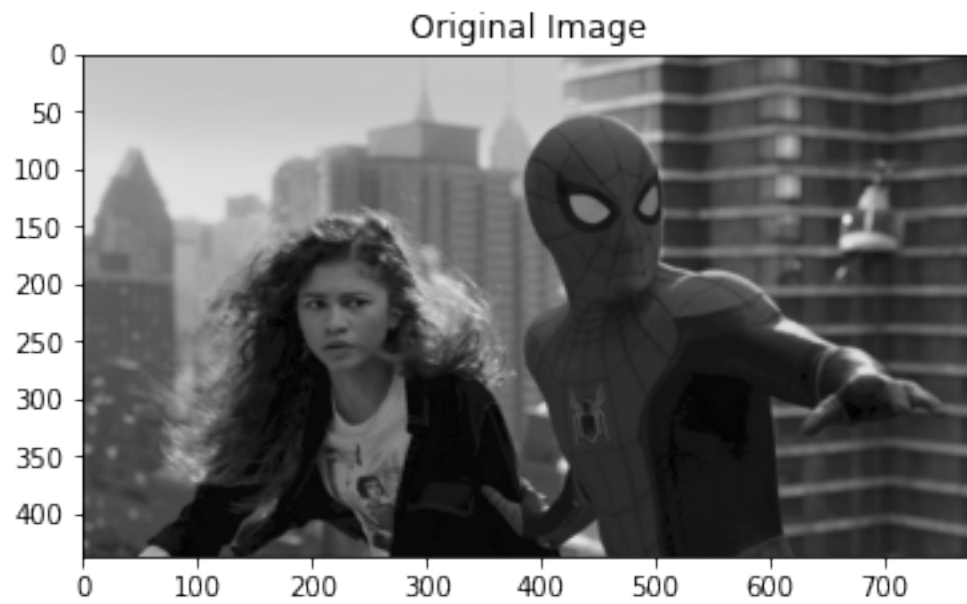
O_image = cv.imread(r'spider.png', cv.IMREAD_GRAYSCALE)
assert O_image is not None
img = cv.cvtColor(O_image, cv.COLOR_BGR2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Original Image')
plt.show()

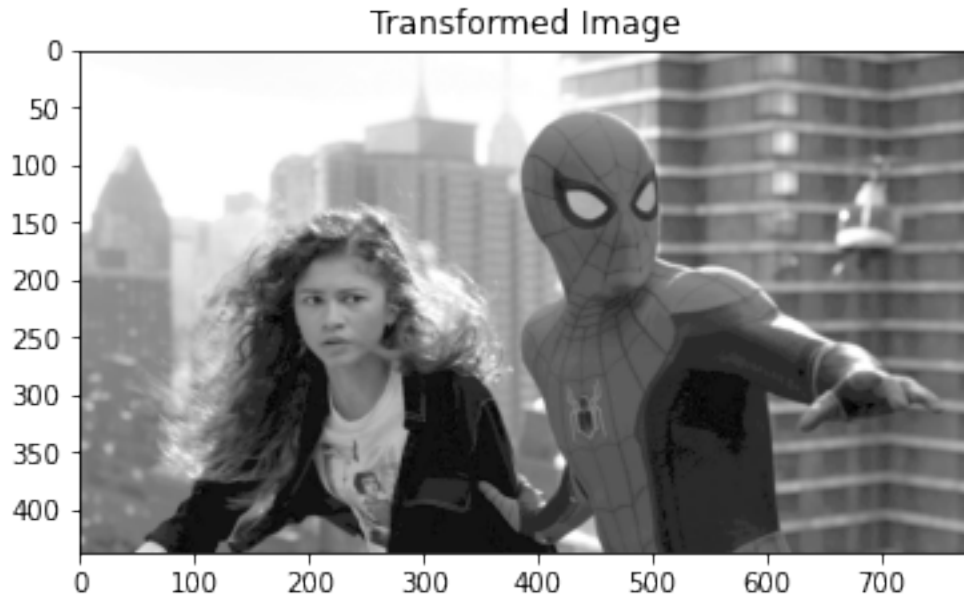
t1 = np.linspace(0, 100, 50)
t2 = np.linspace(100, 255, 150)
t3 = np.linspace(255, 255, 56)

t = np.concatenate((t1, t2, t3), axis = 0).astype(np.uint8)
fig, ax = plt.subplots()
ax.plot(t)
ax.set_aspect('equal')
assert len(t) == 256
Trans_img = cv.LUT(O_image, t)

assert Trans_img is not None
img = cv.cvtColor(Trans_img, cv.COLOR_BGR2RGB)
```

```
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Transformed Image')
plt.show()
```





```
[ ]: #Question 3
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt

O_image = cv.imread(r'shells.tif', cv.IMREAD_GRAYSCALE)
assert O_image is not None
img = cv.cvtColor(O_image, cv.COLOR_BGR2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Original Image')
plt.show()

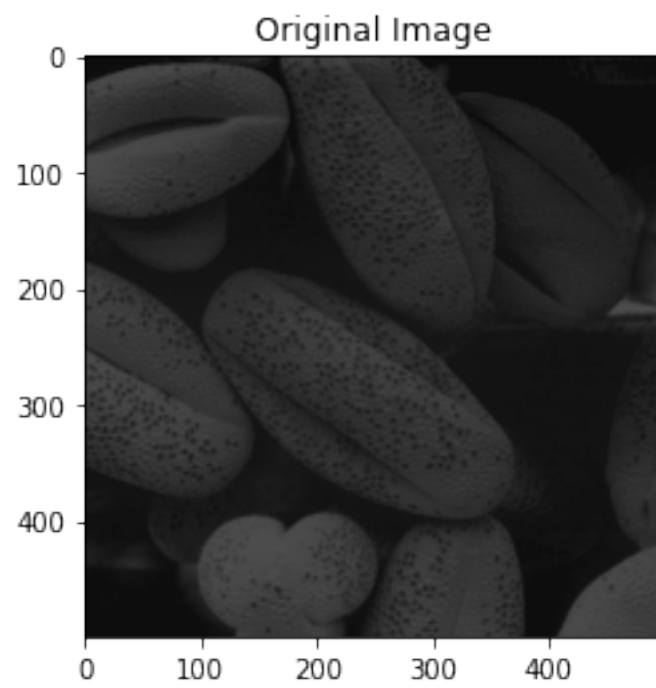
hist_O_image = cv.calcHist([O_image], [0], None, [256], [0,256])
Equalized_image = cv.equalizeHist(O_image)
hist_Equalized_image = cv.calcHist([Equalized_image], [0], None, [256], [0,256])

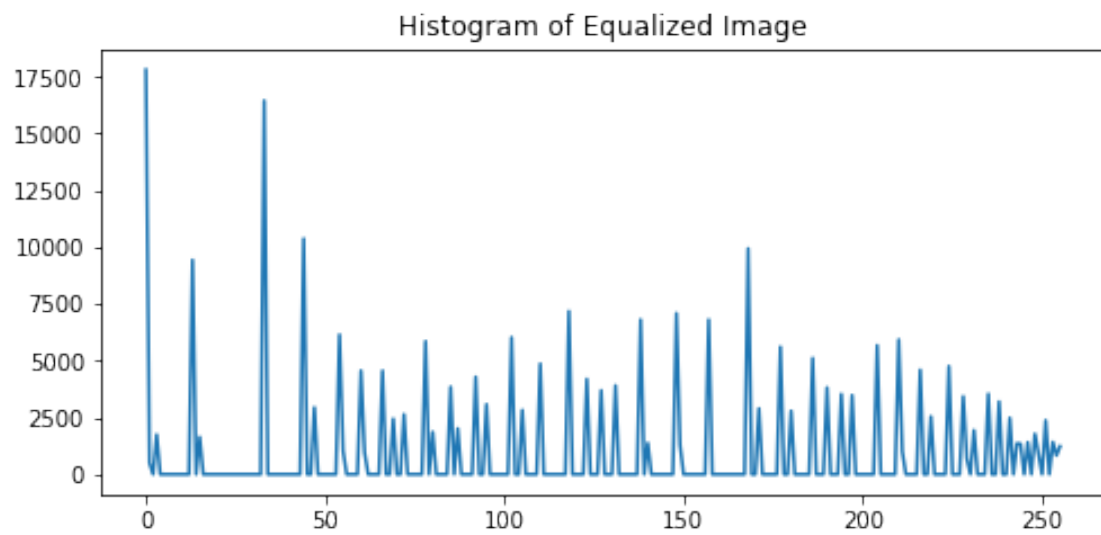
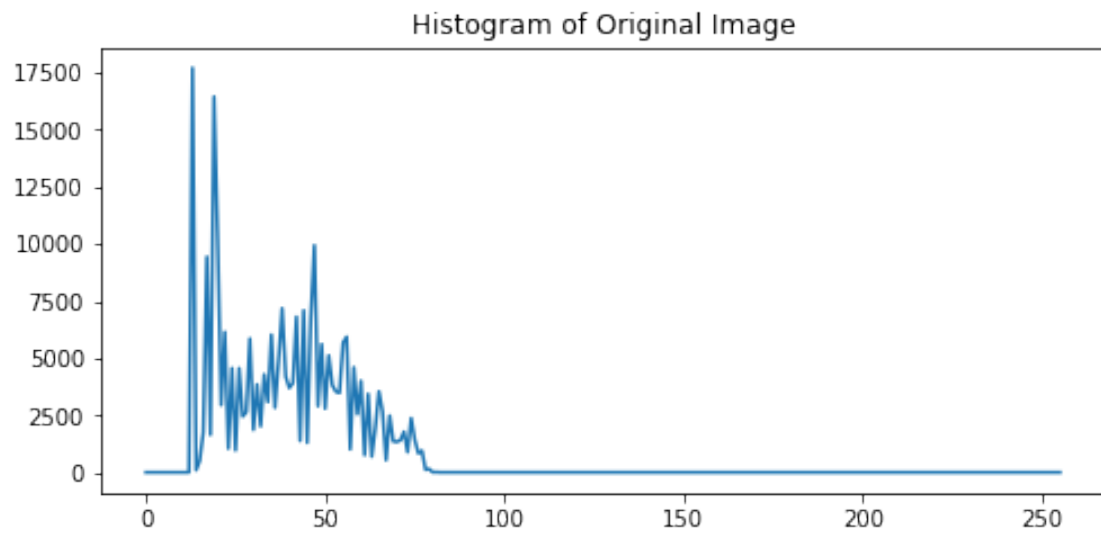
fig, ax = plt.subplots(2, 1, figsize = (8,8))
ax[0].plot(hist_O_image)
ax[0].set_title('Histogram of Original Image')
ax[1].plot(hist_Equalized_image)
ax[1].set_title('Histogram of Equalized Image')

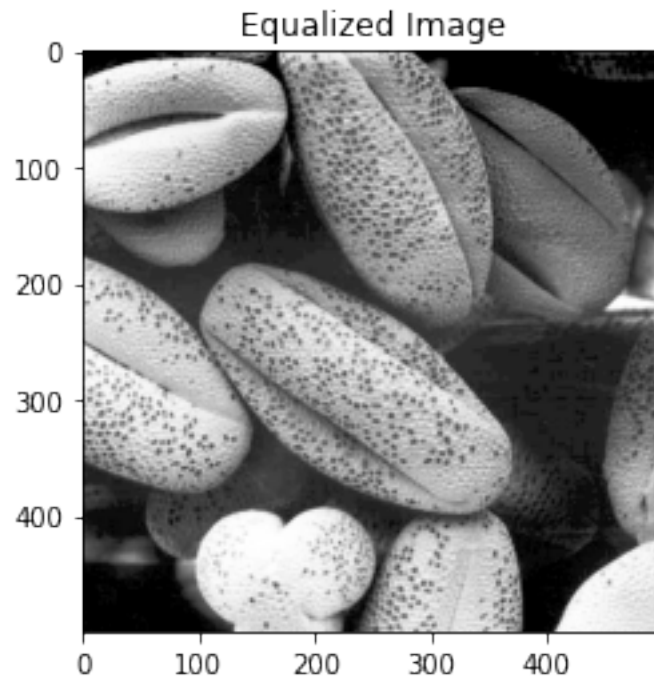
img = cv.cvtColor(Equalized_image, cv.COLOR_BGR2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
```



```
ax.set_title('Equalized Image')  
plt.show()
```







```
[ ]: #Question 4
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt

O_image = cv.imread(r'zion_pass.jpg')
assert O_image is not None
img = cv.cvtColor(O_image, cv.COLOR_BGR2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Original Image')
plt.show()

#Enhancing Saturation
hsv = cv.cvtColor(img, cv.COLOR_BGR2HSV)
hsv[:, :, 1] = hsv[:, :, 1] + 30
hsv[:, :, 1] = np.clip(hsv[:, :, 1], 0, 255)
img = cv.cvtColor(hsv, cv.COLOR_HSV2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Image after Enhancing Saturation')
plt.show()

#Changing Hue
```

```
hsv = cv.cvtColor(img, cv.COLOR_BGR2HSV)
hsv[:, :, 0] = hsv[:, :, 0] + 25
hsv[:, :, 0] = np.clip(hsv[:, :, 0], 0, 255)
img = cv.cvtColor(hsv, cv.COLOR_HSV2RGB)
fig, ax = plt.subplots()
ax.imshow(img)
ax.set_title('Image after Changing Hue')
plt.show()
```

Original Image



Image after Enhancing Saturation

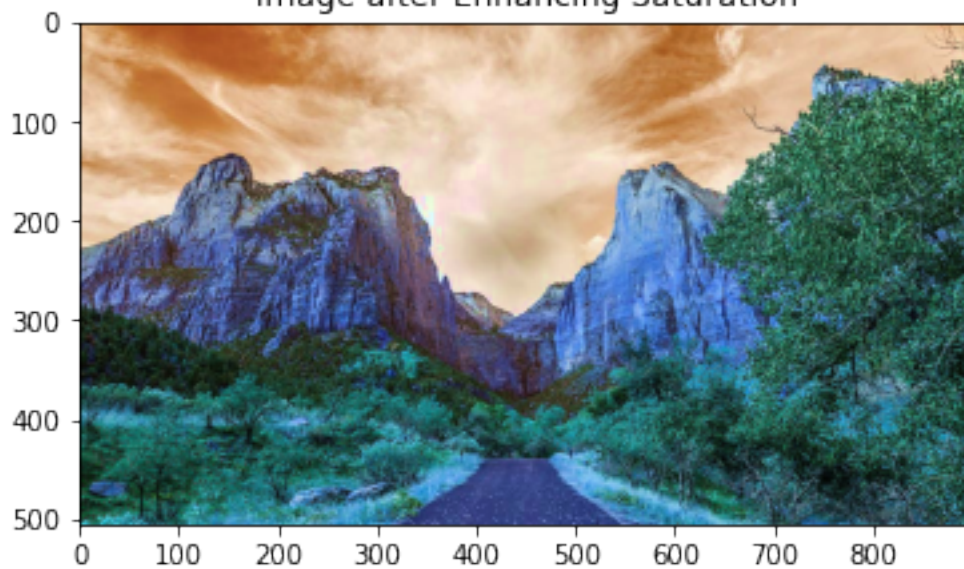


Image after Changing Hue

