EN2550 - Exercise 2 Intensity Transformations

Index No: 190696U

Name: Wijegunawardana C.H.W.

In [3]:

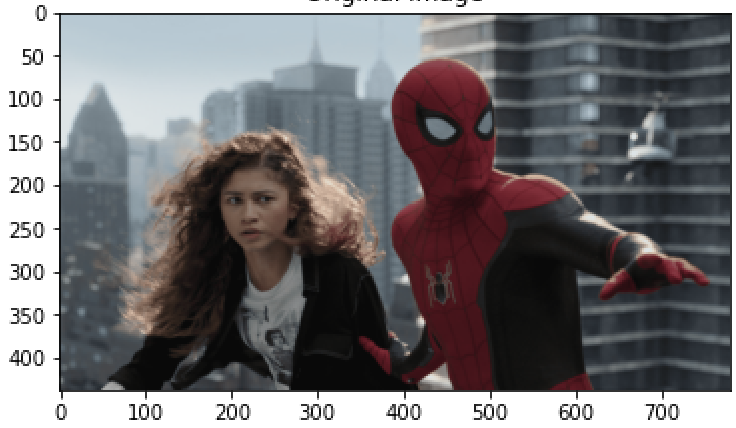
```
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
%matplotlib inline
```

Question 1

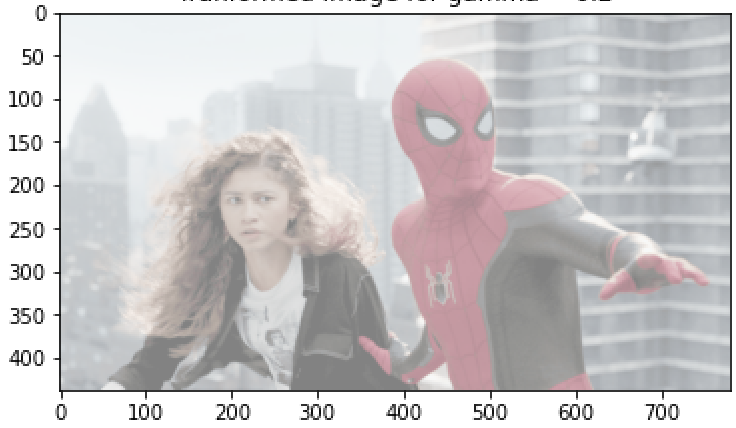
In [15]:

```
# Import the image
img = cv.imread('Images/spider.png', cv.IMREAD_ANYCOLOR)
assert img is not None

# Plot original image
fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img, cv.COLOR_BGR2RGB))
plt.title("Original Image")
plt.show()

gamma = [0.2,0.8,1.2,2]
for g in gamma:
    transform = np.array([(p/255)**g*255 for p in range(256)]).astype(np.uint8)   # Cre
    output = cv.LUT(img,transform)   # Applying the Tranformation

    # Plotting in Matplotlib
    fig,ax = plt.subplots()
    ax.imshow(cv.cvtColor(output, cv.COLOR_BGR2RGB))
    plt.title("Tranformed Image for gamma = " + str(g))
    plt.show()

    # Plotting in openCV Windows
    cv.namedWindow('Image',cv.WINDOW_AUTOSIZE)
    cv.imshow('Image',output)
    cv.waitKey(0)
    cv.destroyAllWindows()
```

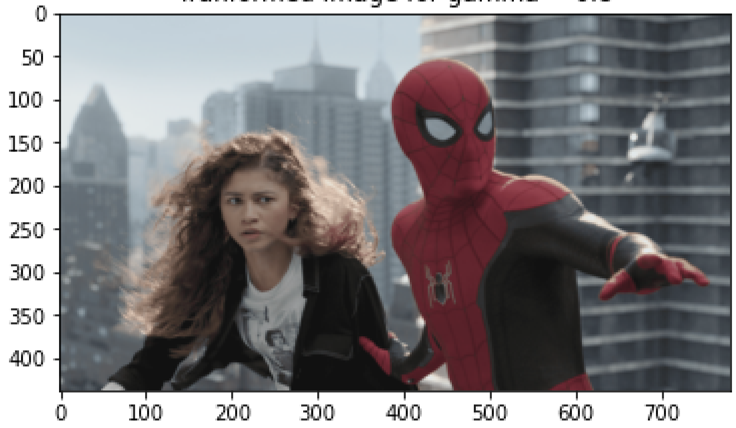
Original Image



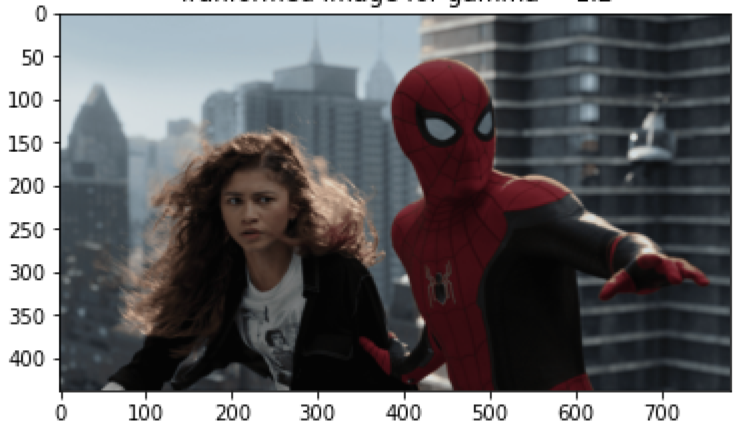
Tranformed Image for gamma = 0.2

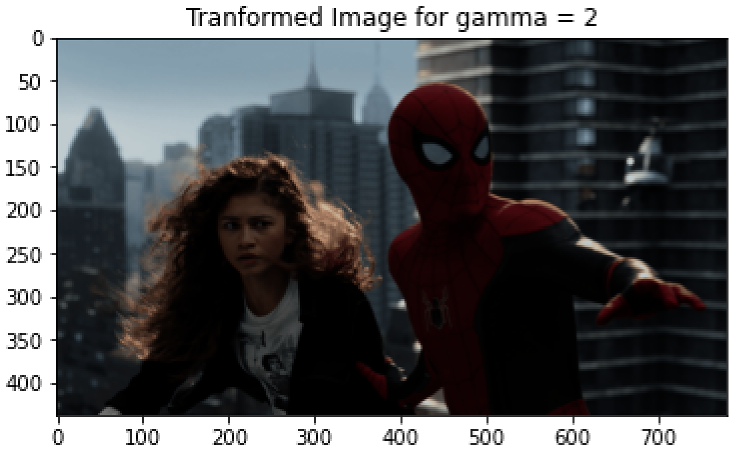


Tranformed Image for gamma = 0.8



Tranformed Image for gamma = 1.2

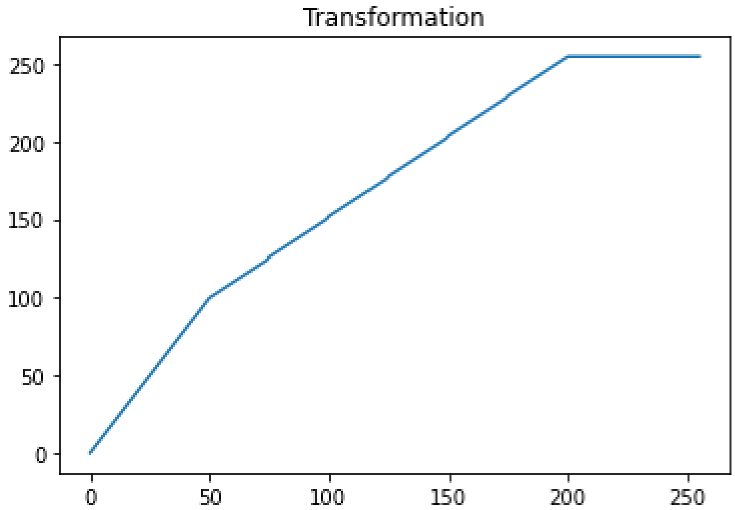


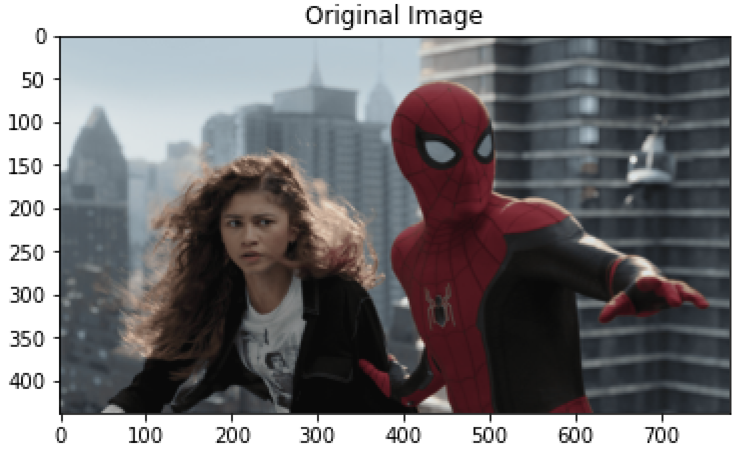


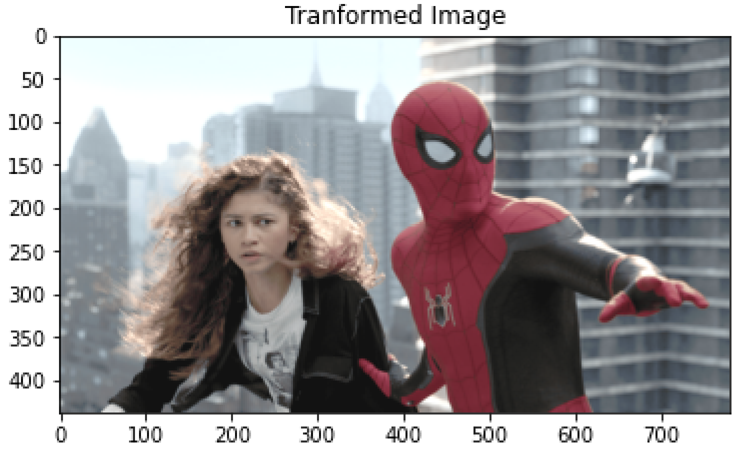
Question 2

In [37]:

```
# Create the Transformation
transform = np.arange(0,256,dtype=np.uint8)
transform[:50] = np.linspace(0,100,50,endpoint=False)
transform[50:200] = np.linspace(100,256,150,endpoint=False)
transform[200:] = 255
#print(transform)

# plotting the transformation
fig,ax = plt.subplots()
ax.plot(transform)
plt.title("Transformation")

# Applying the Transformation
output = cv.LUT(img,transform)

# Plot original image
fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img, cv.COLOR_BGR2RGB))
plt.title("Original Image")
plt.show()

cv.namedWindow('Image',cv.WINDOW_AUTOSIZE)
cv.imshow('Image',img)
cv.waitKey(0)
cv.destroyAllWindows()

# plot transformed image
fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(output, cv.COLOR_BGR2RGB))
plt.title("Tranformed Image")
plt.show()

cv.namedWindow('Image',cv.WINDOW_AUTOSIZE)
cv.imshow('Image',output)
cv.waitKey(0)
cv.destroyAllWindows()
```





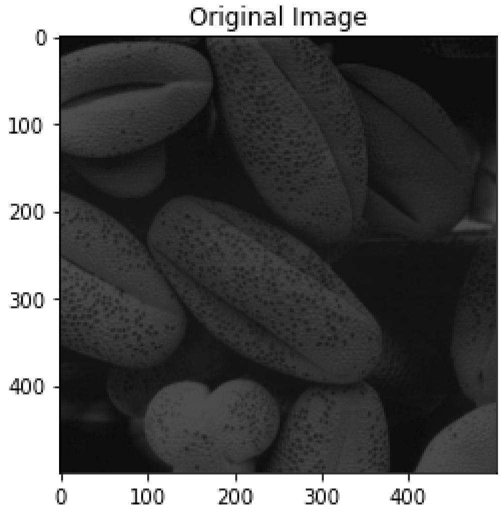


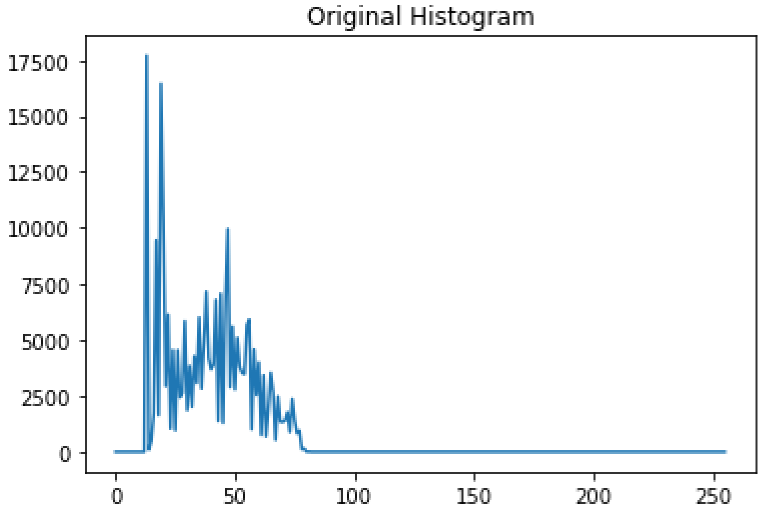
Question 3

In [43]:

```
# Import the image
img = cv.imread('Images/shells.tif', cv.IMREAD_ANYCOLOR)
assert img is not None

fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img, cv.COLOR_BGR2RGB))
plt.title("Original Image")
plt.show()

histogram = cv.calcHist([img],[0],None,[256],[0,256])
```

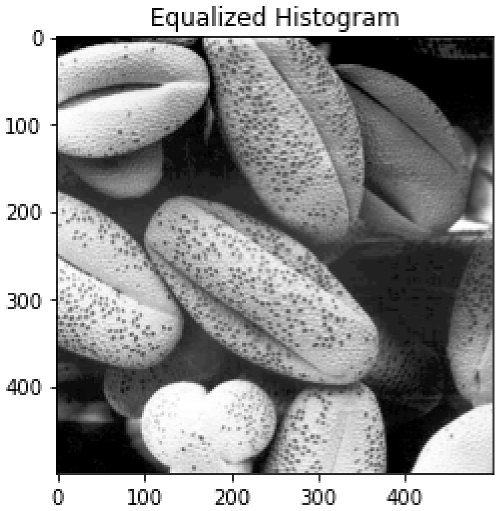
```
fig,ax = plt.subplots()
ax.plot(histogram)
plt.title("Original Histogram")
plt.show()
```

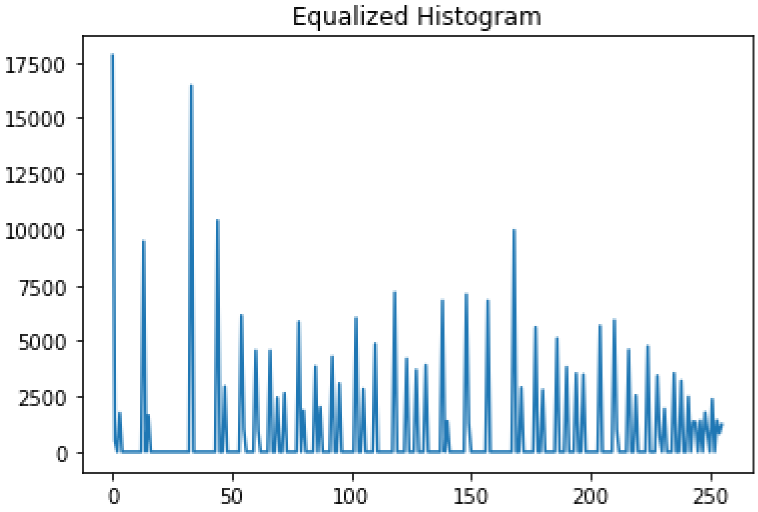




In [51]:

```
# Equalizing
eq_img = cv.equalizeHist(img)
eq_histogram = cv.calcHist([eq_img],[0],None,[256],[0,256])

fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(eq_img, cv.COLOR_BGR2RGB))
plt.title("Equalized Histogram")
plt.show()

fig,ax = plt.subplots()
ax.plot(eq_histogram)
plt.title("Equalized Histogram")
plt.show()
```



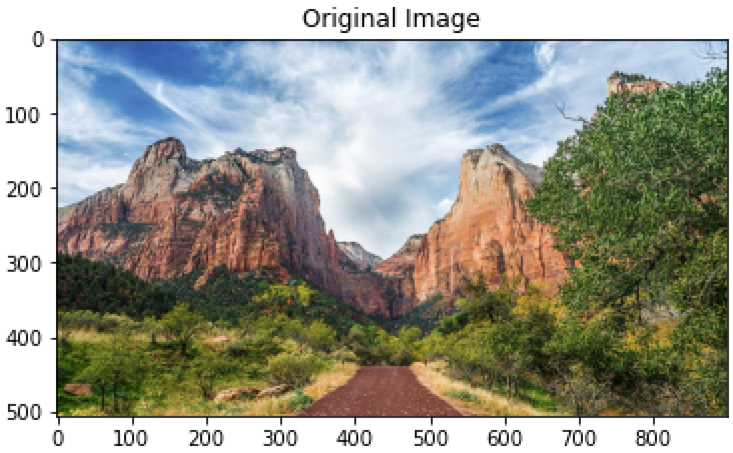


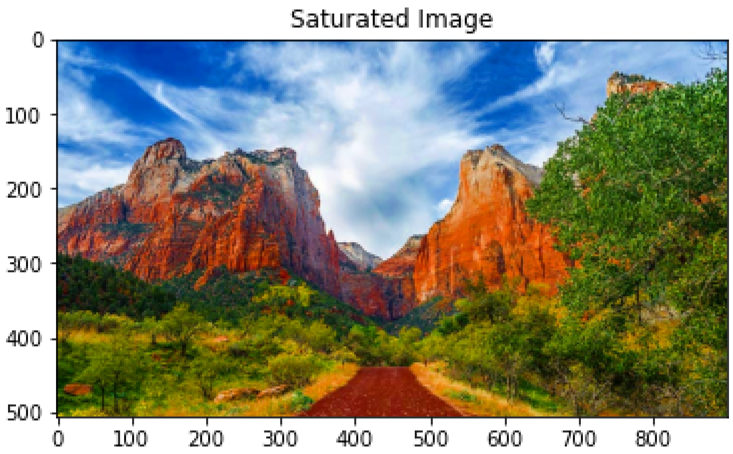
Question 4

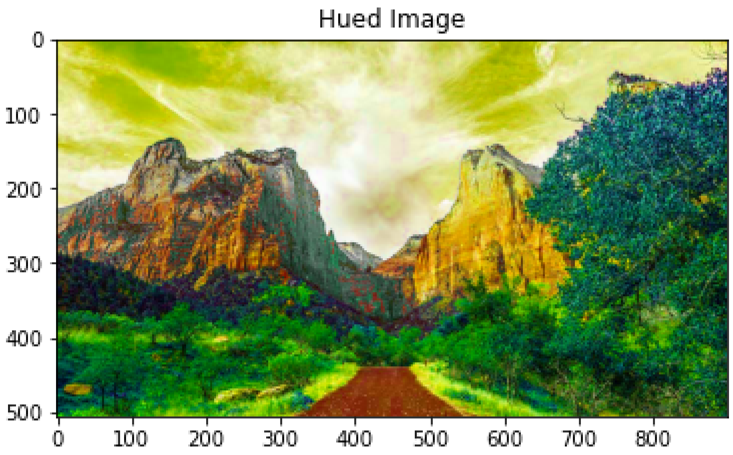
In [57]:

```
# Import the image
img = cv.imread('Images/zion_pass.jpg', cv.IMREAD_ANYCOLOR)
assert img is not None

fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img, cv.COLOR_BGR2RGB))
plt.title("Original Image")
plt.show()

imghsv = cv.cvtColor(img, cv.COLOR_BGR2HSV).astype("float32")
# Saturating
(h, s, v) = cv.split(imghsv)
s = s*2
s = np.clip(s,0,255)
imghsv_sat = cv.merge([h,s,v])
img_sat = cv.cvtColor(imghsv_sat.astype("uint8"), cv.COLOR_HSV2BGR)

# Hue
h = h*2
h = np.clip(h,0,255)
imghsv_hue = cv.merge([h,s,v])
img_hue = cv.cvtColor(imghsv_hue.astype("uint8"), cv.COLOR_HSV2BGR)
```

```
fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img_sat, cv.COLOR_BGR2RGB))
plt.title("Saturated Image")
plt.show()

fig,ax = plt.subplots()
ax.imshow(cv.cvtColor(img_hue, cv.COLOR_BGR2RGB))
plt.title("Hued Image")
plt.show()
```







In [ ]: