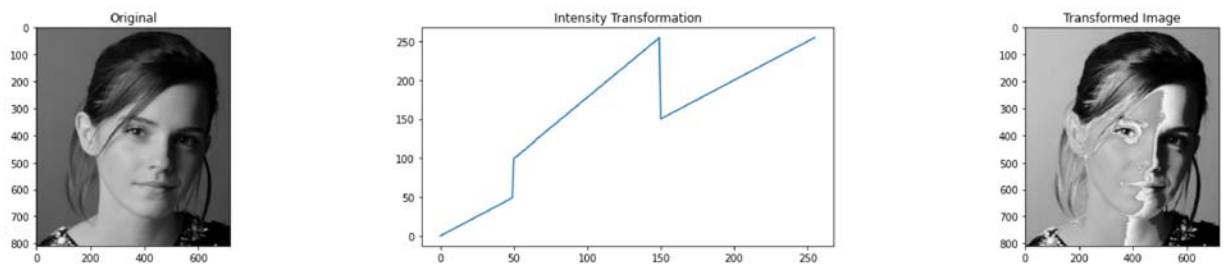


```
In [2]: import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
```

```
In [5]: im=cv.imread('emma_gray.jpg',cv.IMREAD_GRAYSCALE)
assert im is not None
t1=np.linspace(0,50,50)
t2=np.linspace(100,255,100)
t3=np.linspace(150,255,106)
t=np.concatenate((t1,t2,t3),axis=0).astype(np.uint8)
im_1=t[im]
fig,ax=plt.subplots(1,3,figsize=(25,4))
ax[0].imshow(cv.cvtColor(im,cv.COLOR_BGR2RGB))
ax[1].plot(t)
ax[2].imshow(cv.cvtColor(im_1,cv.COLOR_BGR2RGB))
t='Original'
t1='Intensity Transformation'
t2='Transformed Image'
ax[0].set_title(t)
ax[1].set_title(t1)
ax[2].set_title(t2)
```

Out[5]: Text(0.5, 1.0, 'Transformed Image')



```
In [30]: im=cv.imread('brain_proton_density_slice.png',cv.IMREAD_GRAYSCALE)
assert im is not None
fig,ax=plt.subplots(1,1,figsize=(8,4))
ax.imshow(cv.cvtColor(im,cv.COLOR_BGR2RGB))
Text='Original'
ax.set_title(Text)
ax.axis('off')

t1=np.linspace(0,0,212)
t2=np.linspace(255,255,38)
t3=np.linspace(255,255,6)
t=np.concatenate((t1,t2,t3),axis=0).astype(np.uint8)
im_1=t[im]
fig,ax=plt.subplots(2,2,figsize=(25,8))
#ax[0][0].imshow(cv.cvtColor(im,cv.COLOR_BGR2RGB))
ax[0][0].plot(t)
ax[0][1].imshow(cv.cvtColor(im_1,cv.COLOR_BGR2RGB))
#Text='Original'
Text1='Intensity Transformation'
Text2='Transformed Image'
#ax[0][0].set_title(Text)
ax[0][0].set_title(Text1)
ax[0][1].set_title(Text2)

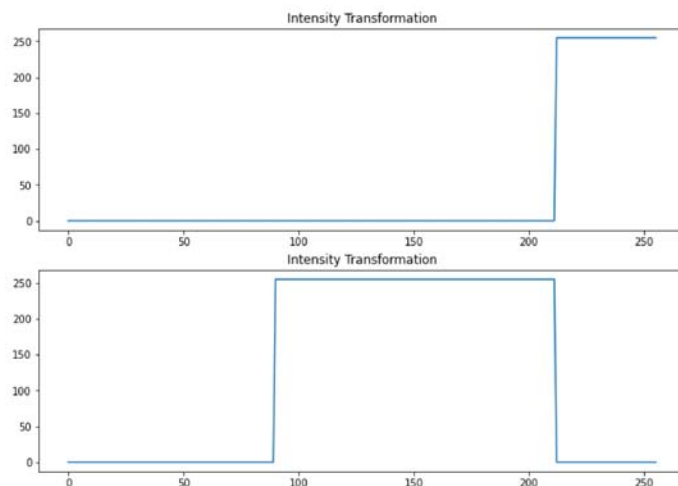
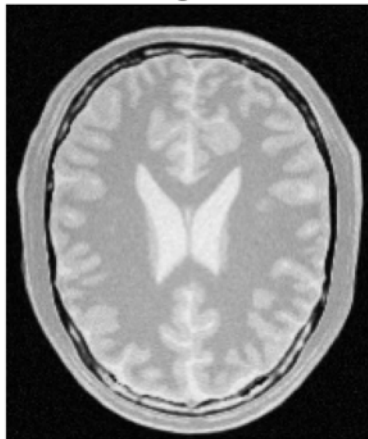
t1=np.linspace(0,0,90)
t2=np.linspace(255,255,122)
t3=np.linspace(0,0,44)
```

```

t_new=np.concatenate((t1,t2,t3),axis=0).astype(np.uint8)
im_2=t_new[im]
#ax[1][0].imshow(cv.cvtColor(im,cv.COLOR_BGR2RGB))
ax[1][0].plot(t_new)
ax[1][1].imshow(cv.cvtColor(im_2,cv.COLOR_BGR2RGB))
#ax[1][0].set_title(Text)
ax[1][0].set_title(Text1)
ax[1][1].set_title(Text2)
for i in range(2):
    ax[i][1].axis('off')

```

Original



Transformed Image



Transformed Image



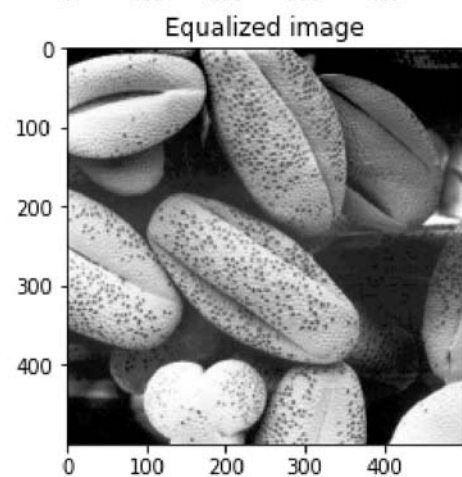
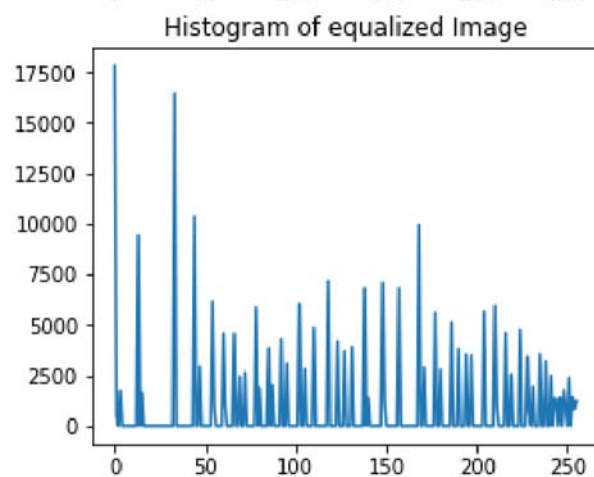
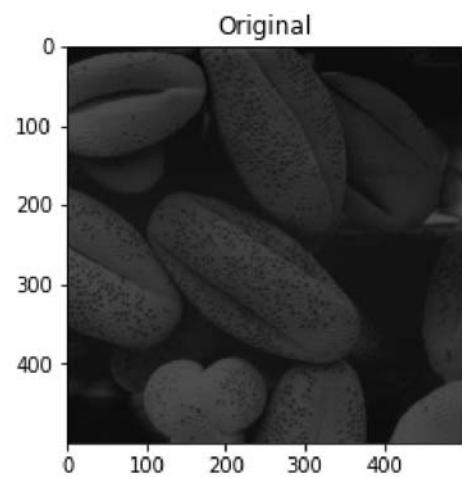
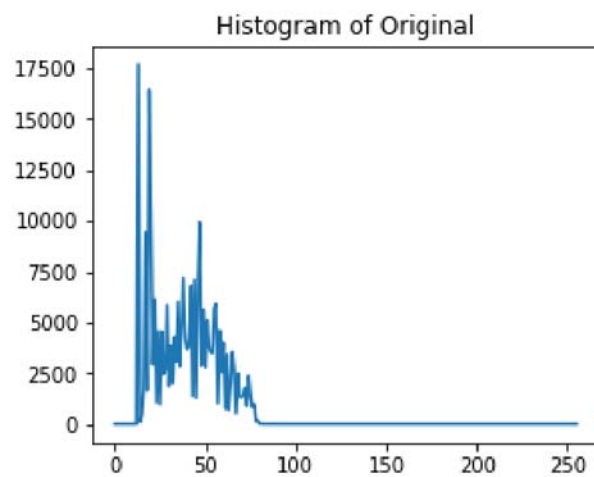
In [32]:

```

f=cv.imread('shells.png',cv.IMREAD_GRAYSCALE)
assert f is not None
hist_f=cv.calcHist([f],[0],None,[256],[0,256])
g=cv.equalizeHist(f)
hist_g=cv.calcHist([g],[0],None,[256],[0,256])
fig,ax=plt.subplots(2,2,figsize=(10,8))
t00='Histogram of Original'
t10='Histogram of equalized Image'
t01='Original'
t11='Equalized image'
ax[0][0].plot(hist_f)
ax[1][0].plot(hist_g)
ax[0][1].imshow(cv.cvtColor(f,cv.COLOR_BGR2RGB))
ax[1][1].imshow(cv.cvtColor(g,cv.COLOR_BGR2RGB))
ax[0][0].set_title(t00)
ax[1][0].set_title(t10)
ax[0][1].set_title(t01)
ax[1][1].set_title(t11)

```

```
Out[32]: Text(0.5, 1.0, 'Equalized image')
```



In []: