الدكتور فاروق العمري

محمد أمين معتصم علي عبابنه

Mohammed Ameen Motasim Ali Ababneh

2017980073

تحليل الصور الرقمية

Exercise #1:

a- Read the image into I

b-

Figure.1: (I) image

c- J similar to image I with the four quadrants shuffled as below

The code below is an process of transforming the image using Python OpenCV

```
J = np.zeros((512,512,3), dtype = np.uint8)
```

```
I1 = I[0:I.shape[1]//2 ,0:I.shape[1]//2]

I2 = I[0:I.shape[1]//2,I.shape[1]//2:I.shape[1]]

I3 = I[I.shape[1]//2:I.shape[1],0:I.shape[1]//2]

I4 = I[I.shape[1]//2:I.shape[1],I.shape[1]//2:I.shape[1]]

J[0:J.shape[1]//2,0:256] = I4

J[0:J.shape[1]//2,J.shape[1]//2:J.shape[1]] = I3
J[J.shape[1]//2:J.shape[1],0:J.shape[1]//2] = I2
J[J.shape[1]//2:J.shape[1],J.shape[1]//2:J.shape[1]] = I1
```

```
plt.imshow(J)

plt.show()
```

d- Display image J.

Figure.2: (J) image

e- Compute the display the histograms of the original image (I) and image (J)

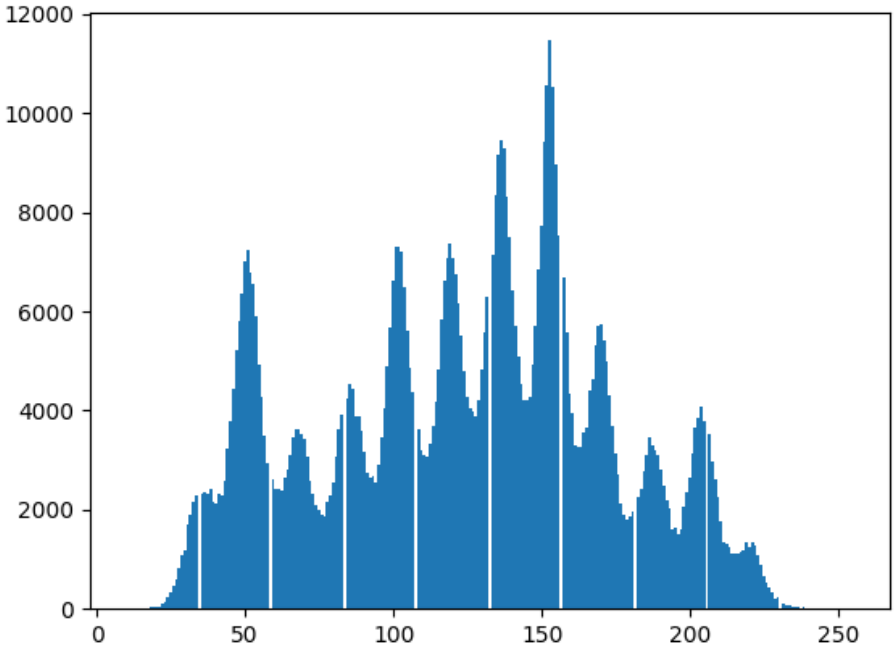

Figure.3: A histogram of the image I

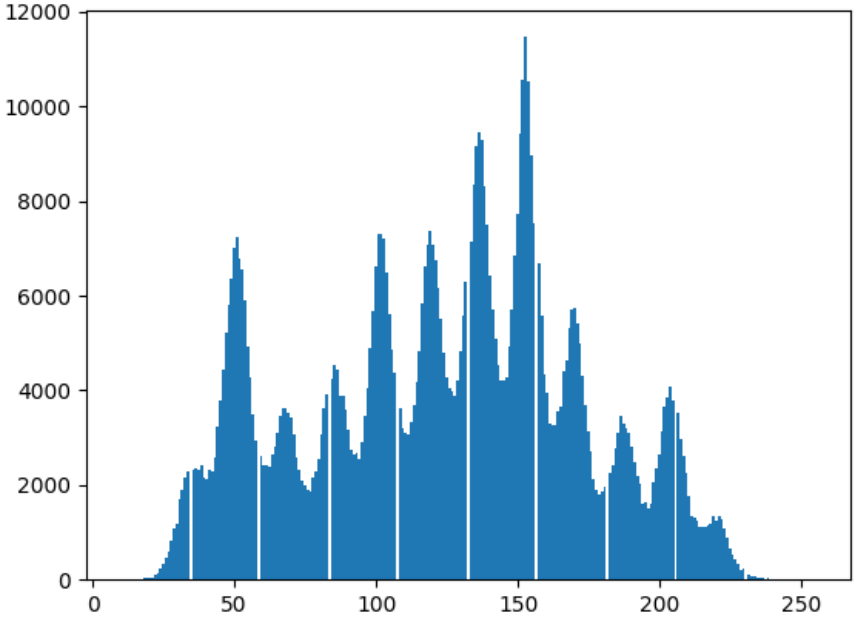

Figure.4: A histogram of the image J

f- What is the difference between the two histograms? And justify your answer.

The histograms of the two image is the same because their was no change in the colors of the modified image just the order of the image, which lead to this result.

Exercise #2:

a- Read the image into K.

Figure.5: The normal image

b- Compute and display image L, which is the negative of K

Figure.6: The negative image

c- Compute and display image R, where R is a rescaled copy of K with Sx = Sy = 2

Figure.7: The resize image

The code was done by using Python with the OpenCV library

This is the code of Exercise #1:

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

I = cv.imread('Photos\lena.jpg') #Reading an image into I

#cv.imwrite('I.png', I) to print the picture to an output

#Splitting the image into four components
I1 = I[0:I.shape[1]//2 ,0:I.shape[1]//2]
I2 = I[0:I.shape[1]//2,I.shape[1]//2:I.shape[1]]
I3 = I[I.shape[1]//2:I.shape[1],0:I.shape[1]//2]
I4 = I[I.shape[1]//2:I.shape[1],I.shape[1]//2:I.shape[1]]

J = np.zeros((512,512,3), dtype = np.uint8) #defining an 512*512 image into J

#putting the four components into J as in part b
J[0:J.shape[1]//2,0:256] = I4
J[0:J.shape[1]//2,J.shape[1]//2:J.shape[1]] = I3
J[J.shape[1]//2:J.shape[1],0:J.shape[1]//2] = I2
J[J.shape[1]//2:J.shape[1],J.shape[1]//2:J.shape[1]] = I1

#cv.imwrite('J.png', J) to print the picture to an output

#Displaying the image
plt.imshow(J)
plt.show()


#Displaying the Histograms for both pictures
plt.hist(I.ravel() , bins = 255)
plt.show()

plt.hist(J.ravel() , bins = 255)
plt.show()
```

This is the code of Exercise #2:

```
import cv2 as cv
import numpy as np

K = cv.imread('Photos\Female.tif') #Reading the image into K

cv.imwrite('K.png',K)

L = ~K #Giving L the negative value of K
cv.imshow('Negative value' , L)  #Display the image
cv.waitKey(0)

#cv.imwrite('L.png',L) to print the picture to an output

#scealing an image by 2
width = int(K.shape[1] * 2)
height = int(K.shape[0] * 2)
dimenstions = (width , height)

R = cv.resize(K, dimenstions , interpolation = cv.INTER_NEAREST) #This for the re
size process giving the function a interploation as nearest which work as the zeo
rth-hold formela

#cv.imwrite('R.png', R) to print the picture to an output

cv.imshow('Rescale by 2' ,R) #Display the image
cv.waitKey(0)
```

Note* The project code including the output images are attached with the File