Question 3 – Report

In section (A) we implement a function to calculate histogram of an image, first thing we import opencv, numpy and matplotlib libraries to use them in our program.

After that we define a function called calc\_histogram with one parameter called original\_image, this parameter represents the path of the image, then we use opencv function called imread to read the image, then we use numpy library to find height and width of the image and store it on variables, then we define an array with 256 elements.

After that we loop in the image values (i represent height value and j represent width value), then we calculate the frequency of each gray level color with this equation:

f[original\_image[i][j]] = f[original\_image[i][j]] + 1

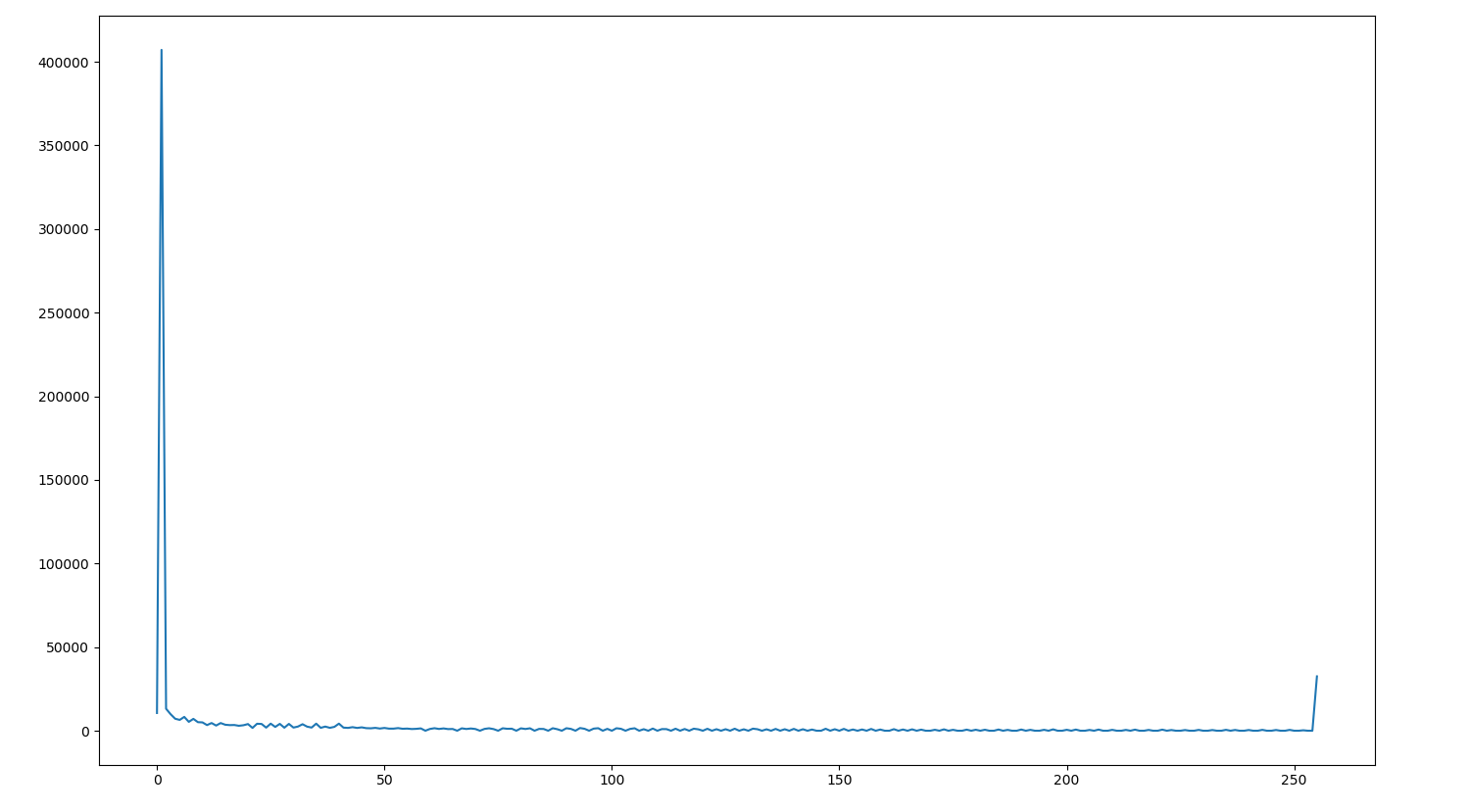
after that we use matplotlib function called figure, then we use another function from this library called plot to draw the histogram result, then show it on the screen.

we test this program on many images, but here we well talk about this function effect on Fig.3.8(a):

this the image we apply calc\_histogram function on it:



and this is the result:



In section (B) we implement a function to equalize histogram of an image and perform this function on Fig.3.8(a), first thing we import opencv, numpy and matplotlib libraries to use them in our program.

After that we define a function called equalize\_hist with one parameter called original\_image, this parameter represents the path of the image, then we use opencv function called imread to read the image, then we use numpy library to find height and width of the image and store it on variables, then we calculate the total number of pixels in the image by multiply width and height and store this value in a variable called total\_number\_of\_pixles.

Then we define an array with 256 elements in it and another array with the same size to store the new gray level values in it.

After that we loop in the image values (i represent height value and j represent width value), then we calculate the frequency of each gray level color with this equation:

f[original\_image[i][j]] = f[original\_image[i][j]] + 1

then we define a new variable called curr, then we loop in f array and calculate the new value of curr the calculate the value of new gray level by this equation:

new\_gray\_level[i] = round((curr \* 255) / total\_number\_of\_pixels)

then we show the old image, after that we calculate the new image value with this equation:

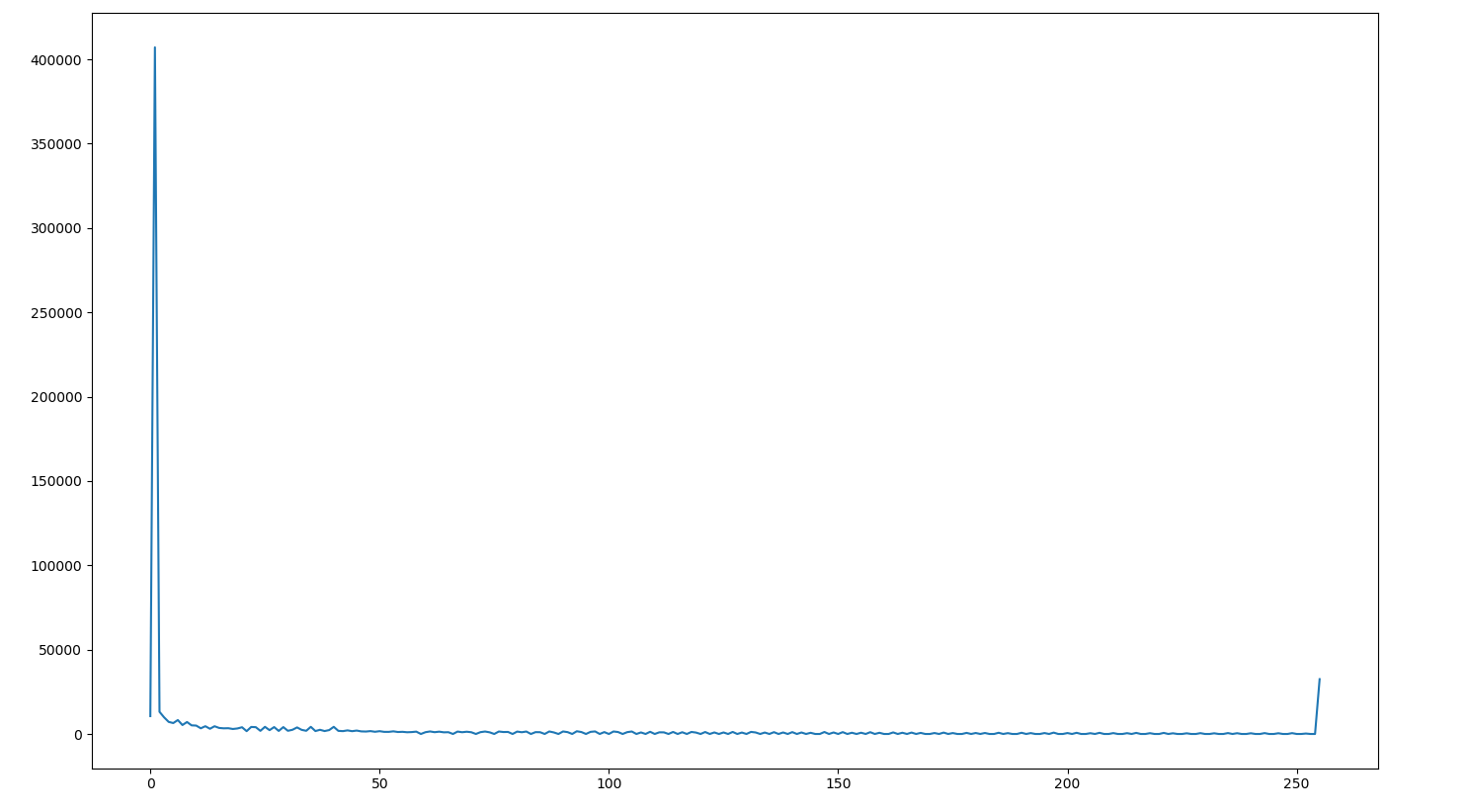
original\_image[i][j] = new\_gray\_level[original\_image[i][j]]

the show the histogram figure of the old and new image and show the two images.

This is the old image this the new image

C:\Users\ME\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Fig0308(a)(fractured_spine).tif 

Old image histogram



New image histogram

