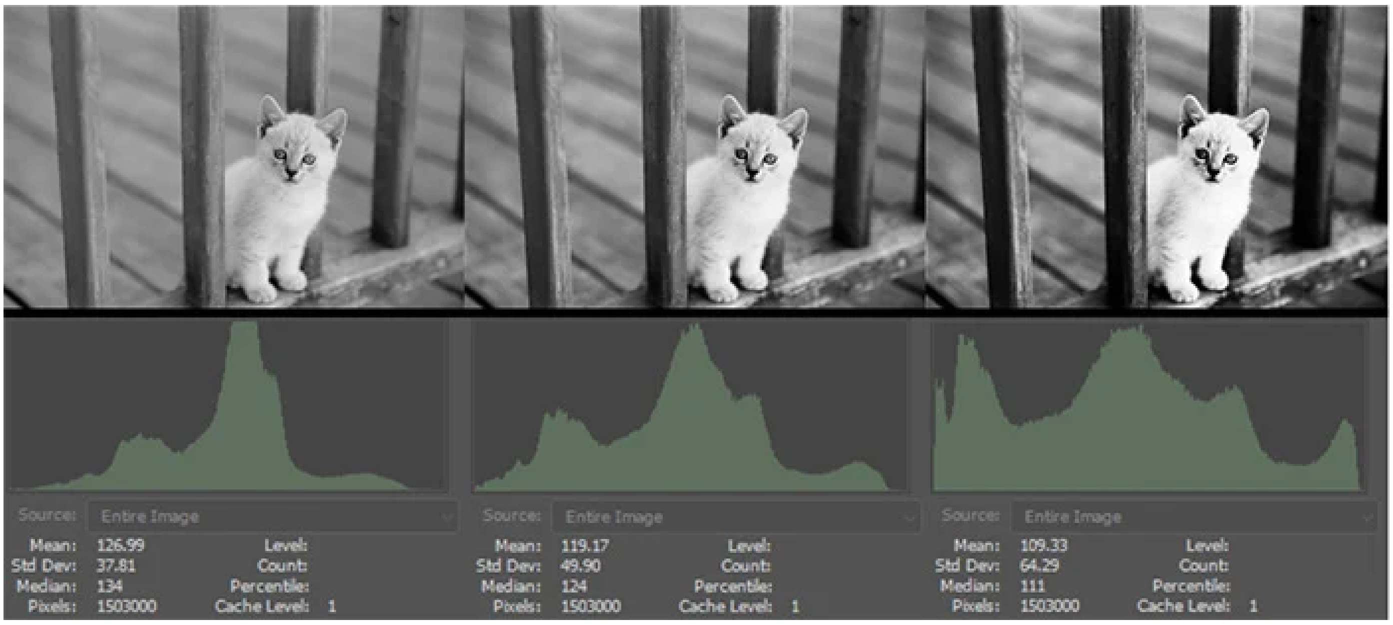
Felix Filipi - 2301877590

Assignment 1

Computer Vision

1. WHAT IS IMAGE HISTOGRAM, AND HOW TO TELL WHETHER AN IMAGE HAVE A GOOD CONTRAST FROM ITS HISTOGRAM.

Image Histogram is a graphical representation that depicts the relationship between the intensity of each pixel in an image. Histograms are represented in the form of bins or bar charts where each bin represents the frequency/intensity of each particular range of values. If we want to **increase contrast** then we need to increase the standard deviation of the frequency histogram, so that it can provide higher contrast because it will make the frequency of each pixel more evenly distributed. As for **reduce contrast** then we need to reduce the standard deviation so that it can provide a lower contrast, because the frequency of the pixels will become more concentrated on a primary peak.

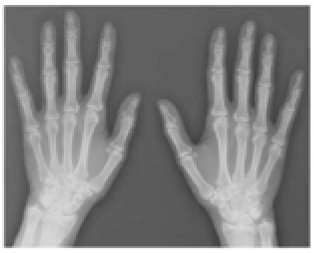


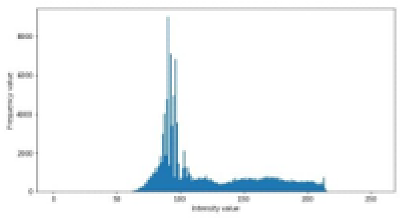
```
Source: Entire Image            Source: Entire Image            Source: Entire Image
  Mean: 126.99      Level:        Mean: 119.17      Level:        Mean: 109.33      Level:
Std Dev: 37.81      Count:      Std Dev: 49.90      Count:      Std Dev: 64.29      Count:
Median: 134    Percentile:      Median: 124    Percentile:      Median: 111    Percentile:
Pixels: 1503000 Cache Level: 1  Pixels: 1503000 Cache Level: 1  Pixels: 1503000 Cache Level: 1
```

The image on the left has a low standard deviation, so the image is more concentrated on the primary peaks causing the image to have low contrast. While the image on the right has a high standard deviation, so it has a more concentrated image on the secondary peak or in other words it is more even than the image on the left, so this causes better contrast.

So according to this explanation, it can be concluded that by making the **pixels more even** (making the ranges and secondary peaks more prominent), we can **increase the contrast**. Meanwhile, by making **more pixels concentrated to the primary peak** then we can **reduce the contrast** of the image.

2. LET SAY YOU HAVE A HISTOGRAM OF A LOW CONTRAST IMAGE AS IT IS SHOWN IN THE FOLLOWING FIGURE, CAN YOU SKETCH A TRANSFORMATION THAT WILL LIKELY IMPROVE THE IMAGE CONTRAST.





As we know, from the image it can be concluded that the contrast of the image is low because it has a high primary peak, which means the contrast is poor. Therefore we need to transform the image into an image with better contrast. This can be done by many transformation techniques, but we will talk about histogram equalization transformations.

Histogram equalization itself is a transformation that works by redistributing the frequency value of each pixel so that it can stretch the contrast to produce an image with better contrast. How this Transformation works is as follows:

1. Make sure the image is in grayscale. 2. Find the hist value (frequency grayscale value of each pixel). 3. Normalize each hist, by:

$$norm = \frac{hist\ array}{total\ pixels}$$

4. Perform cumulative sum (can be seen in code number 3):

$$cs = cumsum(norm)$$

5. Create a transformation with the following formula:

$$round(cs \times L - 1)$$

from:

$$h(v) = round\left(\frac{cdf(v) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L - 1)\right)$$

6. Apply transform to each pixel.

***Note:**

$L - 1$ here means $2^{bit} - 1$

3. WRITE A PROGRAM (IN PYTHON NOTEBOOK) THAT PERFORMS HISTOGRAM EQUALIZATION ON A GRAYSCALE IMAGE. YOUR PROGRAM SHOULD :

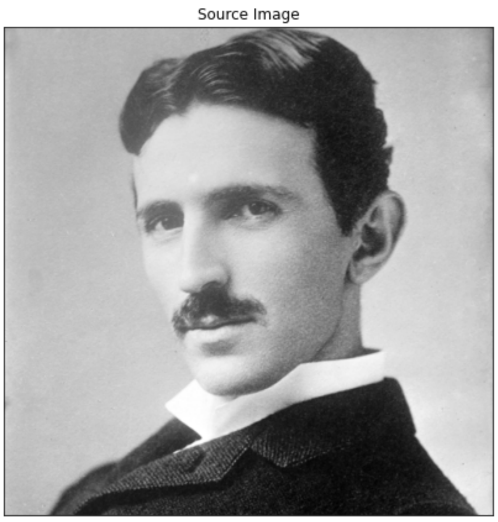
1. COMPUTE THE HISTOGRAM OF THE INPUT IMAGE
2. COMPUTE THE HISTOGRAM OF THE EQUALIZED IMAGE.
3. DISPLAY (AND PRINT) THE ORIGINAL AND EQUALIZED IMAGES AS WELL AS THEIR CORRESPONDING HISTOGRAMS, ALL IN ONE FIGURE.

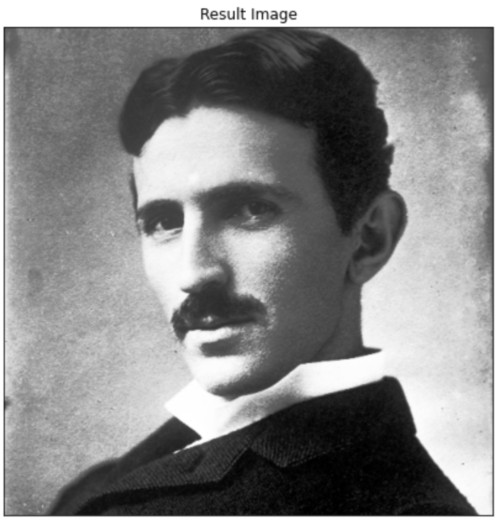
YOU MAY USE OPENCV TO QUICKLY IMPLEMENT THE PROGRAM.

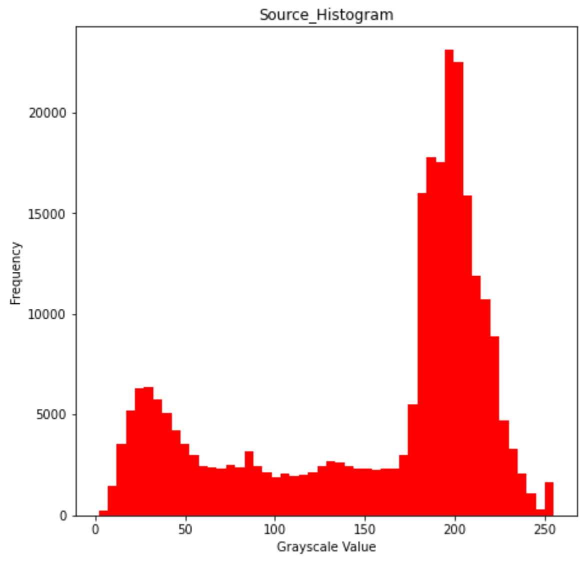
In [1]:

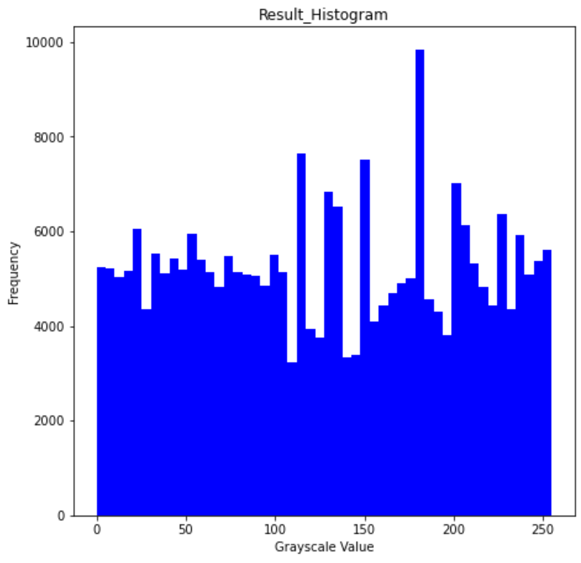
```python
#import lib
import matplotlib.pyplot as plt
import numpy as np
from PIL import Image

img = Image.open('tesla.jpg') #get image
img = img.resize((512,512)) #resize image pixel
img_array = np.asarray(img) #convert to array

hist = np.zeros(256) #grayscale value (between 0-255)

#loop through pixels and append it to the hist
#(find frequency of each grayscale value between 0-255)
for pixel in img_array.flatten():
    hist[pixel]+=1

#normalize each hist array
total_pixels = np.sum(hist) #sum of all hist freq
normalized_hist = hist/total_pixels

#Cumulative summation
iter_hist = iter(normalized_hist)
cum_sum = [next(iter_hist)]
for i in iter_hist:
    cum_sum.append(cum_sum[-1] + i)

cum_sum = np.array(cum_sum)

#Transformation Formula (explained in 2.6)
transform = np.floor(cum_sum * 255).astype(np.uint8) #2^8 -1 = 255 (L-1)

img_arr_list = list(img_array.flatten()) #convert 2D to list of 1D

# transform each pixel
out_img = [transform[p] for p in img_arr_list]

# reshape and return into img_array (for picture)
out_img_array = np.reshape(np.asarray(out_img), img_array.shape)

#plot image
plt.figure(figsize=(16,16))
plt.subplot(2,2,1)
plt.title("Source Image")
plt.yticks([]),plt.xticks([])
plt.imshow(img, cmap='gray')

plt.subplot(2,2,2)
plt.title("Result Image")
plt.yticks([]),plt.xticks([])
plt.imshow(out_img_array, cmap="gray")

#plot histogram
plt.subplot(2,2,3)
plt.title("Source_Histogram")
plt.xlabel("Grayscale Value") #0-255
plt.ylabel("Frequency")
plt.hist(img_array.flatten(),color="red", bins=50)

plt.subplot(2,2,4)
plt.title("Result_Histogram")
plt.xlabel("Grayscale Value")
plt.ylabel("Frequency")
plt.hist(out_img, color="blue",bins=50)

print('Source Image = ',img_array)
print('Result Image = ', out_img_array)
```

```
Source Image =  [[178 178 178 ... 135 123 108]
 [172 178 179 ... 134 122 109]
 [160 178 179 ... 136 117 110]
 ...
 [203 200 198 ... 225 224 221]
 [195 193 195 ... 224 221 219]
 [196 199 201 ... 225 223 220]]
Result Image =  [[100 100 100 ...  78  72  66]
 [ 95 100 101 ...  78  72  66]
 [ 90 100 101 ...  79  70  67]
 ...
 [192 179 169 ... 243 242 237]
 [155 148 155 ... 242 237 233]
 [160 174 183 ... 243 240 235]]
```









As we can see here, the image has a high primary peak which means it doesn't have good contrast. Therefore, we need to perform a transformation so as to produce a better contrast. (bad left, good right)