Histogram-of-an-images

Aim

To obtain a histogram for finding the frequency of pixels in an Image with pixel values ranging from 0 to 255. Also write the code using OpenCV to perform histogram equalization.

Software Required:

Anaconda - Python 3.7

Algorithm:

Step1:

Read the gray and color image using imread()

Step2:

Print the image using imshow().

Step3:

Use calcHist() function to mark the image in graph frequency for gray and color image.

step4:

Use calcHist() function to mark the image in graph frequency for gray and color image.

Step5:

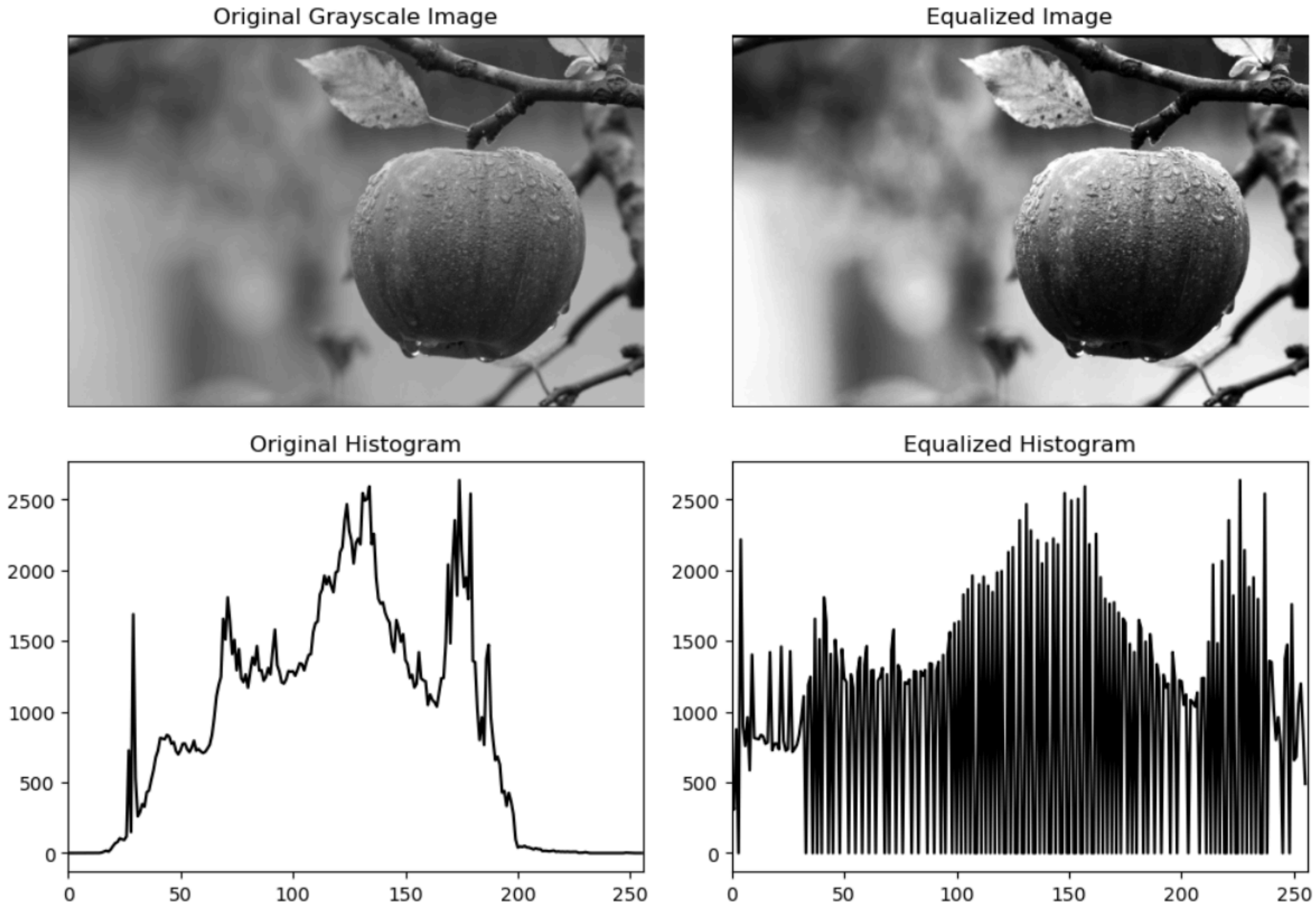
The Histogram of gray scale image and color image is shown.

Program:

```
# Developed By: ALLEN JOVETH P
# Register Number: 212223240007

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

image = cv2.imread('apple.png')
```

```
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

hist_original = cv2.calcHist([gray_image], [0], None, [256], [0, 256])

equalized_image = cv2.equalizeHist(gray_image)

hist_equalized = cv2.calcHist([equalized_image], [0], None, [256], [0, 256])

plt.figure(figsize=(10, 7))

plt.subplot(2, 2, 1)
plt.imshow(gray_image, cmap='gray')
plt.title('Original Grayscale Image')
plt.axis('off')

plt.subplot(2, 2, 2)
plt.imshow(equalized_image, cmap='gray')
plt.title('Equalized Image')
plt.axis('off')

plt.subplot(2, 2, 3)
plt.plot(hist_original, color='black')
plt.title('Original Histogram')
plt.xlim([0, 256])



plt.subplot(2, 2, 4)
plt.plot(hist_equalized, color='black')
plt.title('Equalized Histogram')
plt.xlim([0, 256])

plt.tight_layout()
plt.show()
```

Output:



Result:

Thus the histogram for finding the frequency of pixels in an image with pixel values ranging from 0 to 255 is obtained. Also,histogram equalization is done for the gray scale image using OpenCV.