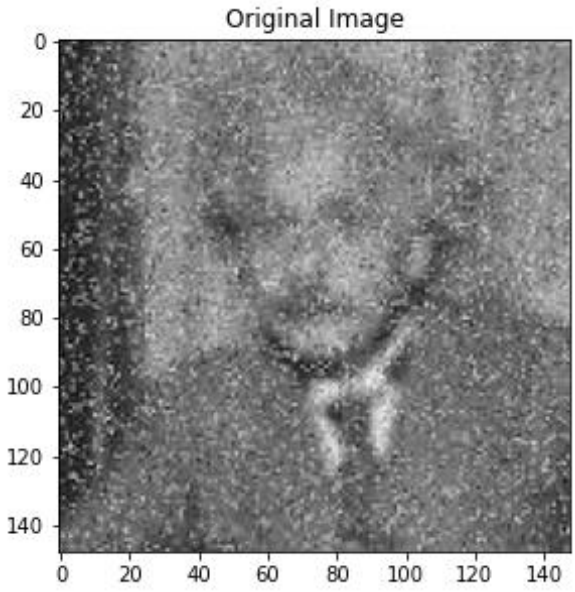
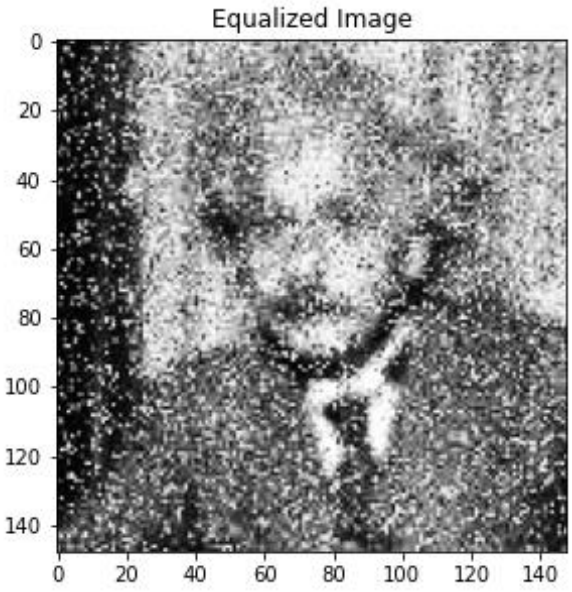
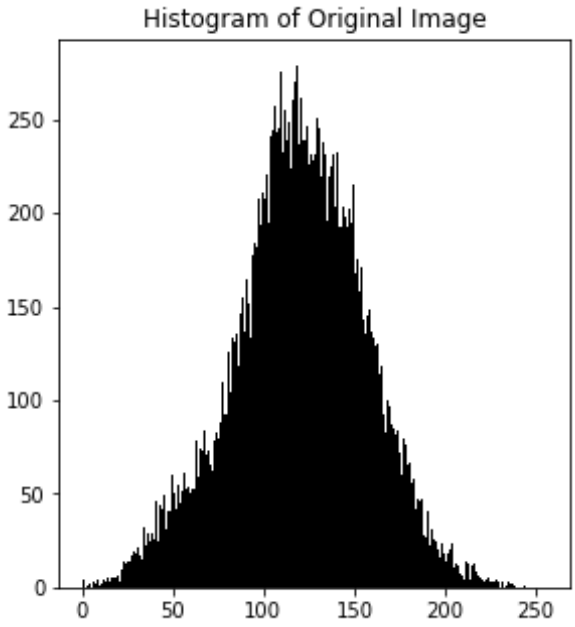
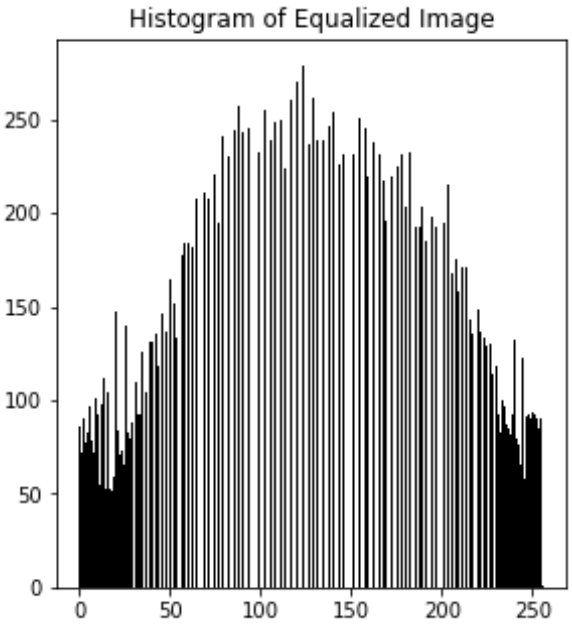
```
# -*- coding: utf-8 -*-
"""
Created on Thu Aug 29 21:34:55 2024
@author: dhanu
"""
import cv2
import numpy as np
import matplotlib.pyplot as plt

def histogram_equalization(image):
    rows, cols = image.shape

    histogram = np.zeros(256)
    for i in range(rows):
        for j in range(cols):
            histogram[image[i, j]] += 1

    cdf = np.zeros(256)
    cdf[0] = histogram[0]
    for i in range(1, 256):
        cdf[i] = cdf[i - 1] + histogram[i]



    cdf_min = np.min(cdf[np.nonzero(cdf)])
    cdf_max = cdf[-1]
    cdf_normalized = (cdf - cdf_min) * 255 / (cdf_max - cdf_min)
    cdf_normalized = cdf_normalized.astype('uint8')

    equalized_image = np.zeros_like(image)
    for i in range(rows):
        for j in range(cols):
```

```
            equalized_image[i, j] = cdf_normalized[image[i, j]]

    return equalized_image

image = cv2.imread('C:\\Users\\dhanu\\OneDrive\\Desktop\\download.jpeg',
cv2.IMREAD_GRAYSCALE)

equalized_image = histogram_equalization(image)

plt.figure(figsize=(10, 5))
plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(image, cmap='gray')
plt.subplot(1, 2, 2)
plt.title('Equalized Image')
plt.imshow(equalized_image, cmap='gray')
plt.show()

plt.figure(figsize=(10, 5))
plt.subplot(1, 2, 1)
plt.title('Histogram of Original Image')
plt.hist(image.flatten(), bins=256, range=[0, 256], color='black')
plt.subplot(1, 2, 2)
plt.title('Histogram of Equalized Image')
plt.hist(equalized_image.flatten(), bins=256, range=[0, 256], color='black')
plt.show()
```


Original Image
0
20
40
60
80
100
120
140
0
20
40
60
80
100
120
140


Equalized Image
0
20
40
60
80
100
120
140
0
20
40
60
80
100
120
140


Histogram of Original Image
0
50
100
150
200
250
0
50
100
150
200
250


Histogram of Equalized Image
0
50
100
150
200
250
0
50
100
150
200
250