1. **Digital Negative of Image Using Histogram equalization.**

**Background:**

Negative, photographic image that reproduces the bright portions of the photographed subject as dark and the dark parts as light areas. A color image stores 3 different channels. They are red, green and blue. That’s why color images are also known as RGB images. So, we need a negative transformation of an image then we need to invert these 3 channels.

**Language:** Python

**Source Code:**

import cv2

import matplotlib.pyplot as plt

# Read an image

img\_bgr = cv2.imread('./Image\_Processing/photo2.jpg',3)

plt.imshow(img\_bgr)

plt.show()

# Histogram plotting of original image

color = ('b', 'g', 'r')

for i, col in enumerate(color):

histr = cv2.calcHist([img\_bgr],

[i], None,

[256],

[0, 256])

plt.plot(histr, color = col)

# Limit X - axis to 256

plt.xlim([0, 256])

plt.show()

cv2.waitKey(0)

cv2.destroyAllWindows()

# Negate the original image

img\_neg = 1 - img\_bgr

plt.imshow(img\_neg)

plt.show()

# Histogram plotting of

# negative transformed image

color = ('b', 'g', 'r')

for i, col in enumerate(color):

histr = cv2.calcHist([img\_neg],

[i], None,

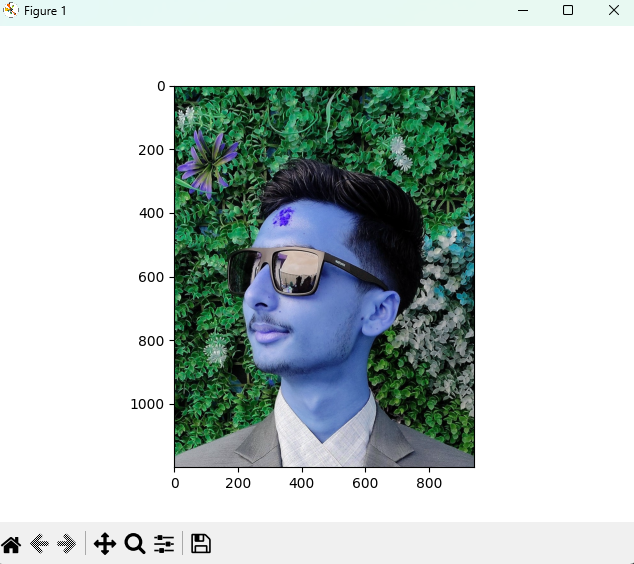
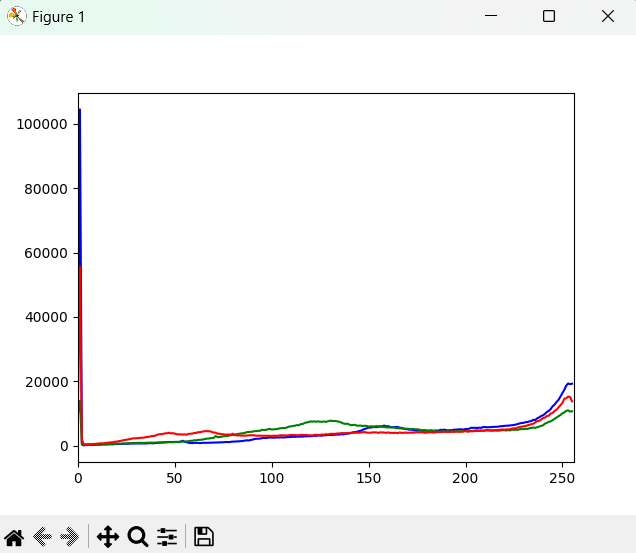
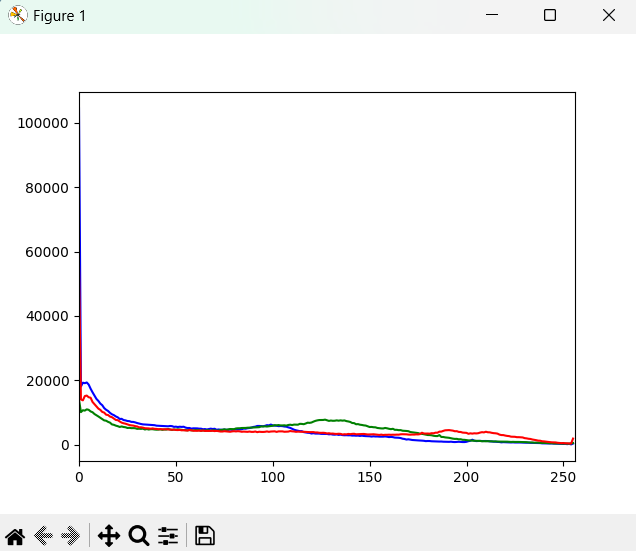
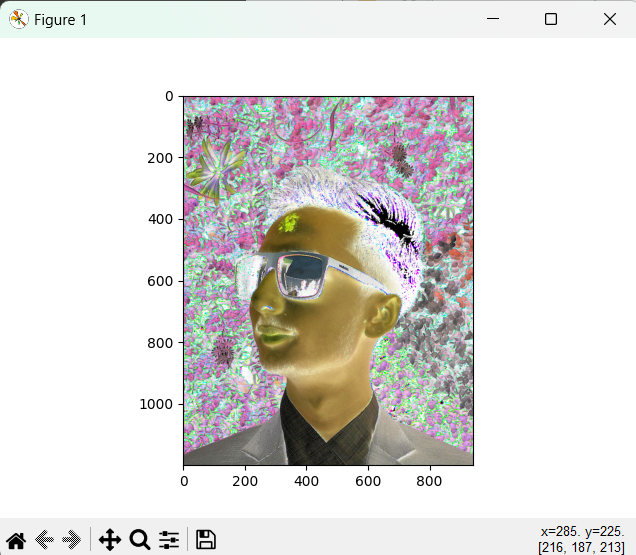
[256],

[0, 256])

plt.plot(histr, color = col)

plt.xlim([0, 256])

plt.show()

**Output:**

**Conclusion:**

In this lab work, we have learned how to obtained the Equalized Histogram from the original Histogram using OpenCV and Matplot library.