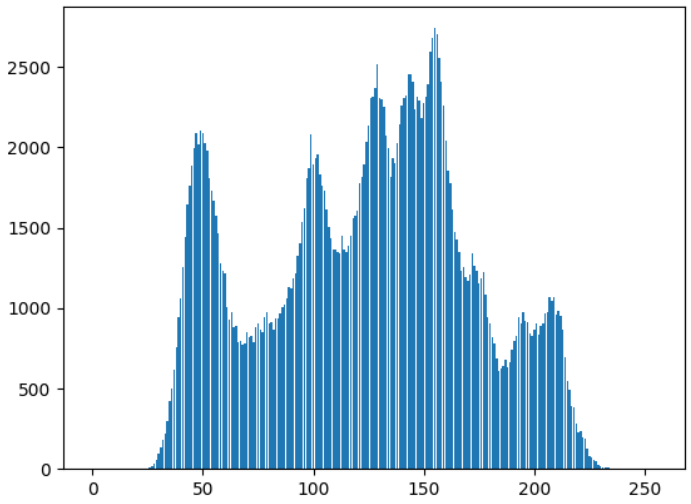
B06902042　劉愷爲

Computer Vision HW3

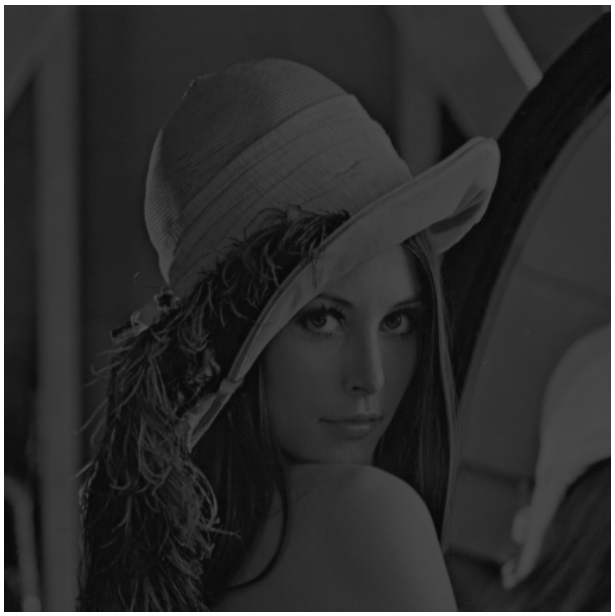
1. original image and its histogram

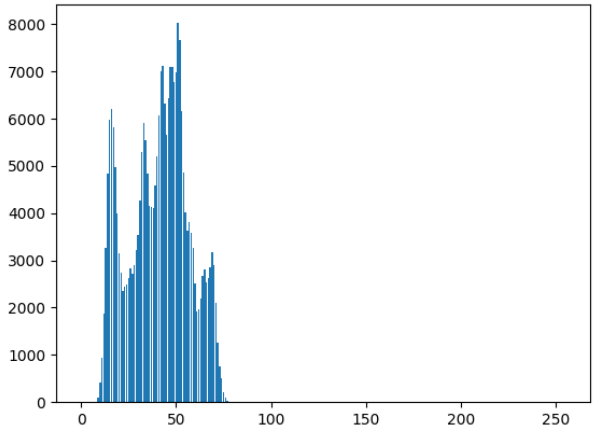
```
def histogram(img):
    ans = np.zeros(256, np.int)
    for x in range(img.shape[1]):
        for y in range(img.shape[0]):
            ans[int(img[x][y])] += 1
    fig = plt.figure()
    plt.bar(range(len(ans)), ans)
```



2. image with intensity divided by 3 and its histogram

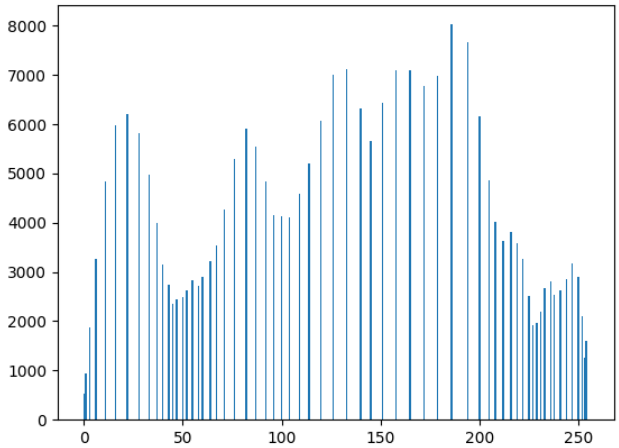
```
img_divided = img_gray / 3
cv2.imwrite("divided3.bmp", img_divided)
histogram(img_divided)
plt.savefig("divided3_histogram.png")
```





3.image after applying histogram equalization to (2) and its histogram

```
def equalization(img):
    new_img = np.zeros((img.shape), np.int)
    eq = np.zeros(256, np.int)
    num = np.zeros(256, np.int)
    for x in range (img.shape[1]):
        for y in range (img.shape[0]):
            num[int(img[x, y])] += 1
    cdf = 0
    for x in range (256):
        cdf += num[x]
        eq[x] = (255 * cdf) // (512 * 512)
    for x in range (img.shape[1]):
        for y in range(img.shape[0]):
            new_img[x, y] = eq[int(img[x, y])]
    return new_img
```



2