Computer Vision HW2

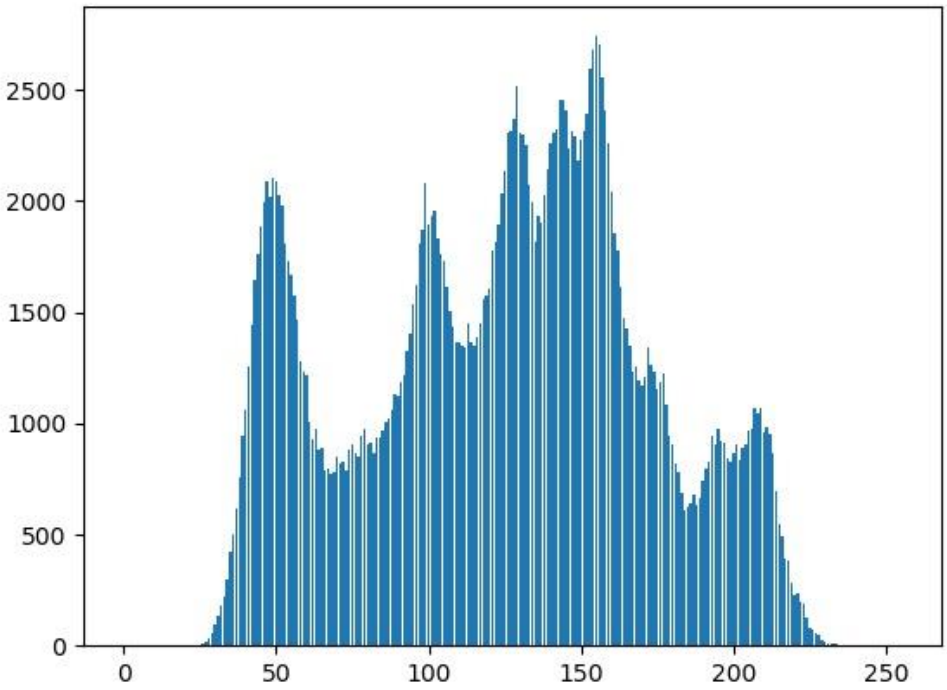
R08922156 黃劍韜

(a) a binary image (threshold at 128)

```python
# (a) A binary image (threshold at 128)
def binary_image(image,width,height,T):
    img_result = image.copy()
    for i in range(width):
        for j in range(height):
            if(img_result[i,j]>=128):
                img_result[i,j] = 255
            else:
                img_result[i,j] = 0
    if(T==True):
        save_image('Binary_image',img_result)
    return img_result
```

第一題的部分，我是去讀取每一個灰階 Image 的 pixel，如果 pixel 的亮度值大於 128 門檻值，那麼我們將這個 pixel 設定為 255，反之設定為 0。即可得到二值化的圖片。

(b) a histogram




```python
# (b) A histogram
def img_histogram(image,width,height):
    Y = np.zeros(256)
    X = [i for i in range(256)]
    for i in range(width):
        for j in range(height):
            Y[image[i,j]] += 1
    plt.bar(X,Y)
    plt.savefig('Histogram_image.jpg')
```

第二題的部分，我是創建一個空矩陣，然後去讀取每一個 pixel，檢查每一個 pixel 對應的亮度，並在矩陣上面累加，最後用 pyplot 函數畫出來 histogram 的部分。

(c) connected components(regions with + at centroid, bounding box)

```python
# (c) connected components (regions with + at centroid,  bounding box)
def connect_compoent(image,width,height):
    img1 = binary_image(image,width,height,False)
    img_result = img1.copy()
    label_num = 1
    label_dic = {}

    # First-Pass
    for i in range(width):
        for j in range(height):
            if(img_result[i,j]==255):
                if(i==0):
                    if(j==0 or img_result[i,j-1]==0):
                        img_result[i,j] = label_num
                        label_num = label_num + 1
                    else:
                        img_result[i,j] = img_result[i,j-1]
                else:
                    if(img_result[i-1,j]==0):
                        if(j==0 or img_result[i,j-1]==0):
                            img_result[i,j] = label_num
                            label_num = label_num + 1
                        else:
                            img_result[i,j] = img_result[i,j-1]
                    else:
                        if(j==0 or img_result[i,j-1]==0):
                            img_result[i,j] = img_result[i-1,j]
                        else:
```

```python
                        if(img_result[i-1,j]<=img_result[i,j-1]):
                            img_result[i,j] = img_result[i-1,j]
                        else:
                            img_result[i,j] = img_result[i,j-1]

                        if(img_result[i-1,j]!=img_result[i,j-1]):
                            if(len(label_dic.keys())==0):
                                cand_list = [ img_result[i,j-1],img_result[i-1,j] ]
                                r_list = [np.max(cand_list)]
                                label_dic[np.min(cand_list)] = r_list
                            else:
                                cand_list = [ img_result[i,j-1],img_result[i-1,j] ]
                                if(np.min(cand_list) in label_dic.keys()):
                                    label_dic[np.min(cand_list)] = list(set(label_dic[np.min(cand_list)] + [np.max(cand_list)]))
                                else:
                                    cand_list = [ img_result[i,j-1],img_result[i-1,j] ]
                                    r_list = [np.max(cand_list)]
                                    label_dic[np.min(cand_list)] = r_list
```

```python
    # print(label_dic)
    sorted_dic = {}
    for i in sorted(label_dic):
        sorted_dic[i] = label_dic[i]

    # # Second-Pass
    count_dic = {}
    for i in range(width):
        for j in range(height):
            for k in sorted_dic.keys():
                if(img_result[i,j] in sorted_dic[k]):
                    img_result[i,j] = k
                    if(img_result[i,j] not in count_dic.keys()):
                        count_dic[img_result[i,j]] = 1
                    else:
                        count_dic[img_result[i,j]] = count_dic[img_result[i,j]] + 1
                    break

    for k in count_dic.keys():
        if(count_dic[k]>=500):
            p1,p2 = get_xy(img_result,width,height,k)
            # print(p1,p2)
            draw = cv2.rectangle(image, p1, p2, (0, 0, 0), 2)
            draw = cv2.circle(draw, (int((p1[0]+p2[0])/2), int((p1[1]+p2[1])/2)), 5,(0, 0, 255), -1)

    show_img(draw)
```

第三題的部分，我是採用上課提到的 Two-Pass 的方法，先掃描一次去給每一個 pixel 標記 label，我採用的是**四連通**的方法，所以每次只用看左邊一個 pixel 以及上面一個 pixel。然後當左邊和上面 pixel 的值不同的時候，取最小的作為標籤，並記錄下連通的 pixel。在第二次的時候，重新掃描一次，然後用更小的 label 代替原本的標籤。最後過來數量小於 500 的標籤然後畫出 bounding box 以及重心。但是我中間步驟有出錯，所以雖然找出五個但是卻是錯的。希望助教之後可以提供參考解答，感謝。