

## 電腦視覺 HW2

R10922176 陳冠穎

(a) Binary-image

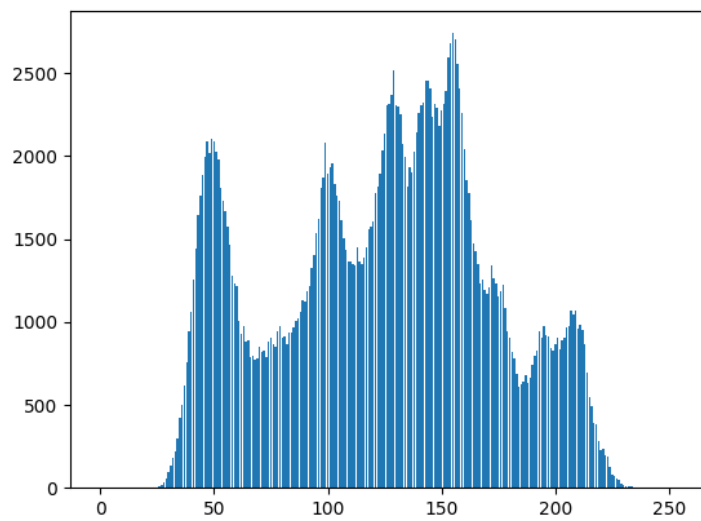


說明:

```
for i in range(img.shape[0]):  
    for j in range(img.shape[1]):  
        if img[i][j] >= 128:  
            retimg[i][j] = 255  
        else:  
            retimg[i][j] = 0
```

如果原像素值大於等於 128 就設定為 255 全白，否則為 0 全黑。

## (b) Histogram



說明：

```
hist_gray = [float(0)] * 256

for row in range(img.shape[0]):
    for col in range(img.shape[1]):
        value = int(img[row][col])
        hist_gray[value] += 1
```

使用一個 List 紀錄每個灰階值出現的次數，之後使用 matplotlib

畫出直方圖。

(c) Connected components



說明：

```
# init each 1-pixel a unique label
for i in range(labeled_img.shape[0]):
    for j in range(labeled_img.shape[1]):
        if bin_img[i][j] != 0:
            labeled_img[i][j] = label
            label += 1

flag1 = True
flag2 = True
while(flag1 or flag2):
    flag1 = top_down_pass(labeled_img)
    flag2 = bottom_up_pass(labeled_img)

return labeled_img
```

我使用講義上的 iterative 演算法來找出所有 connected component。

```
dict[labeled[i][j]][0] = min(dict[labeled[i][j]][0], i)
dict[labeled[i][j]][1] = max(dict[labeled[i][j]][1], j)
dict[labeled[i][j]][2] = max(dict[labeled[i][j]][2], i)
dict[labeled[i][j]][3] = min(dict[labeled[i][j]][3], j)
dict[labeled[i][j]][4] += 1
dict[labeled[i][j]][5] += i
dict[labeled[i][j]][6] += j
```

再來掃過一遍 labeled 後的圖片找出每個 connected component 的

最上、右、下、左邊界，以及 component 的面積，以及每組 row

與 col 的加總，再用這些資訊畫出 bounding box，circle at

centroid。