

Computer Vision HW2 Report

B07902045 資工四 呂紹齊

(a)

```
for i in range(r):  
    for j in range(c):  
        if(img[i][j] >= 128):  
            img[i][j] = 255  
        else:  
            img[i][j] = 0
```

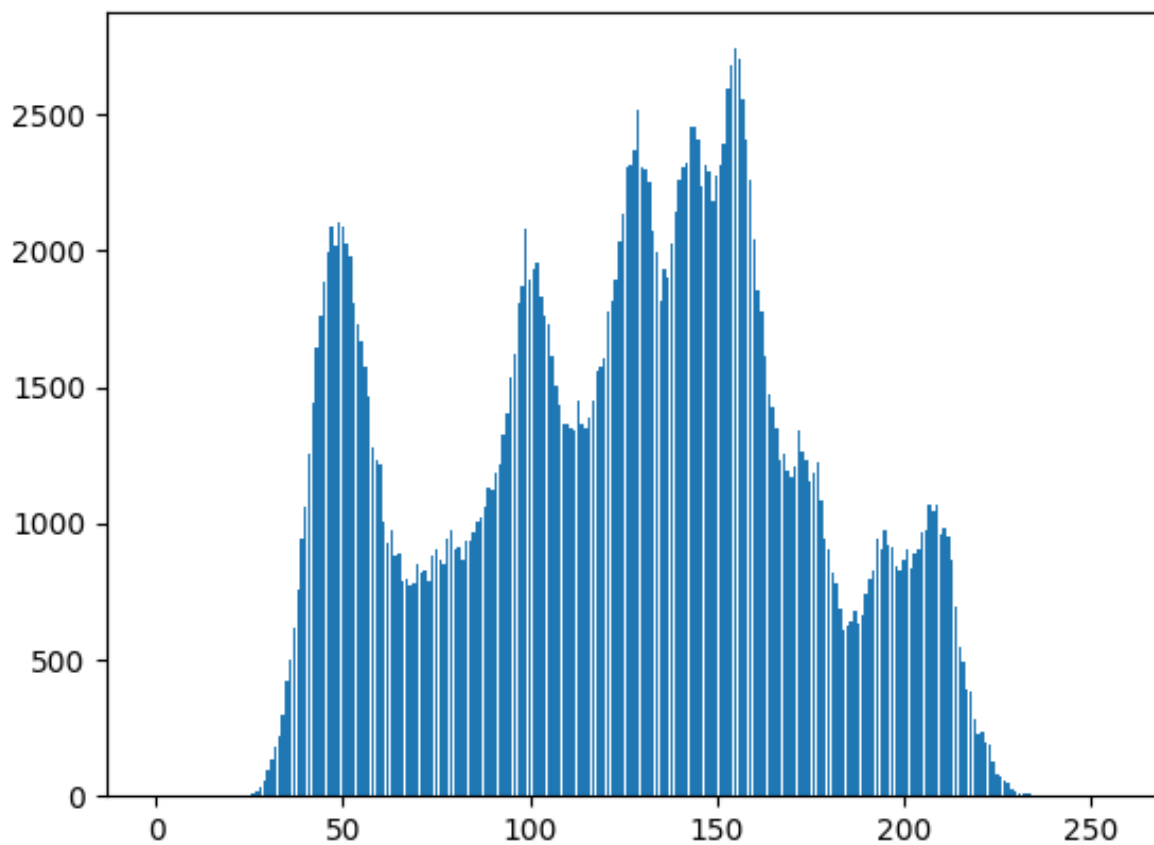


(b)

```
y = np.zeros(256, dtype = np.int)

for i in range(r):
    for j in range(c):
        y[im.getpixel((i,j))] += 1

x= np.arange(256)
plt.bar(x,y)
plt.savefig("histogram", format = "png")
```



(c)

Step 1 (Find components)

利用 iterative 演算法。首先給予所有 binary-1 的 pixel 一個 unique label。Top-down 先掃一次整張圖得到第一階段 4-connected 的 region 分佈圖。接著再 Bottom-up 掃一次圖，得到完整的 conncted component 分佈圖。pseudocode 如上課投影片：



2.3.3 An Iterative Algorithm

procedure Iterate;

1. “Initialization of each 1-pixel to a unique label”

```
for L := 1 to NLines do
  for P := 1 to NPIXELS do
    if I(L,P) = 1
      then LABEL (L,P) := NEWLABEL( )
    else LABEL(L,P) := 0
    end for
  end for;
```

2. “Iteration of top-down followed by bottom-up passes”

repeat

 “Top-down pass”

 CHANGE := false;

```
for L := 1 to NLines do
  for P := 1 to NPIXELS do
    if LABEL(L,P) <> 0 then
      begin
        M := MIN(LABELS(NEIGHBORS((L,P)) ∪ (L,P)));
        if M <> LABEL(L,P)
          then CHANGE := true;
        LABEL(L,P) := M
      end
    end for
  end for;
```

 “Bottom-up pass”

```
for L := NLines to 1 by -1 do
  for P := NPIXELS to 1 by -1 do
    if LABEL(L,P) <> 0 then
      begin
        M := MIN(LABELS(NEIGHBORS((L,P)) ∪ (L,P)));
        if M <> LABEL(L,P)
          then CHANGE := true;
        LABEL(L,P) := M
      end
    end for
  end for
until CHANGE := false
end Iterate
```

36

Step 2 (Find boundary)

掃過整個圖，過程中持續更新每個region的 top, left, down, right, count

```
""labelList: [ label_number, top, left, down, right, count ] ""
labelList = []

for i in range(1,row+1,1):
  for j in range(1,col+1,1):
    if img_table[i][j] != 0:
      if len(labelList) == 0:
        labelList.append([img_table[i][j],i-1,j-1,i-1,j-1,1])
      else:
        isExist = False
        for k in range(len(labelList)):
```

```
if img_table[i][j] == labelList[k][0]:
    isExist = True
    if labelList[k][1] > i-1:
        labelList[k][1] = i-1
    if labelList[k][2] > j-1:
        labelList[k][2] = j-1
    if labelList[k][3] < i-1:
        labelList[k][3] = i-1
    if labelList[k][4] < j-1:
        labelList[k][4] = j-1
    labelList[k][5] += 1

    break
if isExist == False:
    labelList.append([img_table[i][j], i-1, j-1, i-1, j-1, 1])
```

接著挑出個數不少於 500 個的 label，總共有5個。使用ImageDraw的rectangle並將合法的五個region之(top,down,left,right) 當作參數傳入便可畫出boundary。

Step 3 (Find centroid)

掃過圖一遍，將相同region的每個 pixel 座標相加，最後再除上區域大小就是centroid位置。用ImageDraw的line在centroid附近畫兩條紅色短線形成十字。

