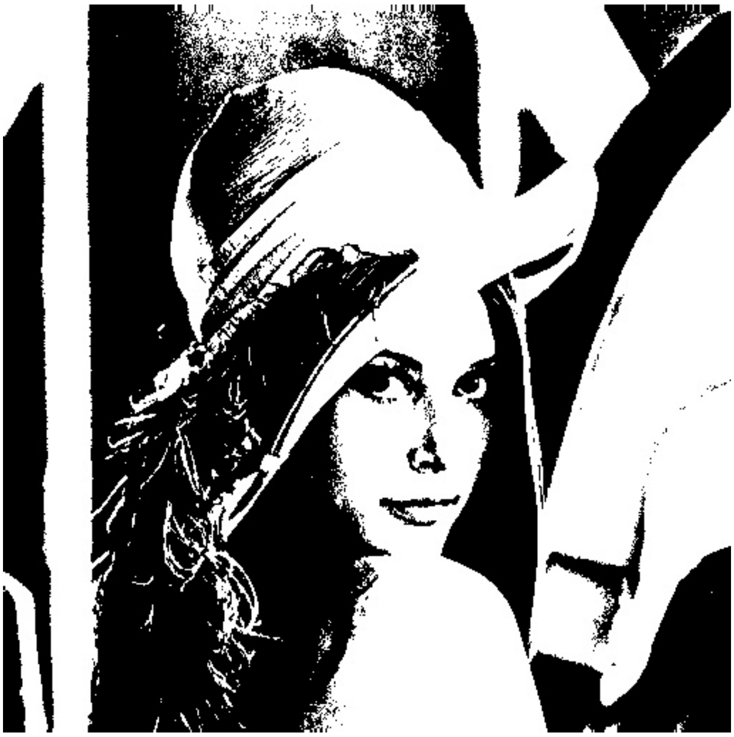
Computer Vision 2018Fall
HW02
電信所碩一 謝硯澤 R07942115

使用環境說明：

```
#macOS Majave 10.14
#Python 3.7.0
#openCV 3.4.2
#PIL 5.2.0
#matplotlib 3.0.0
```

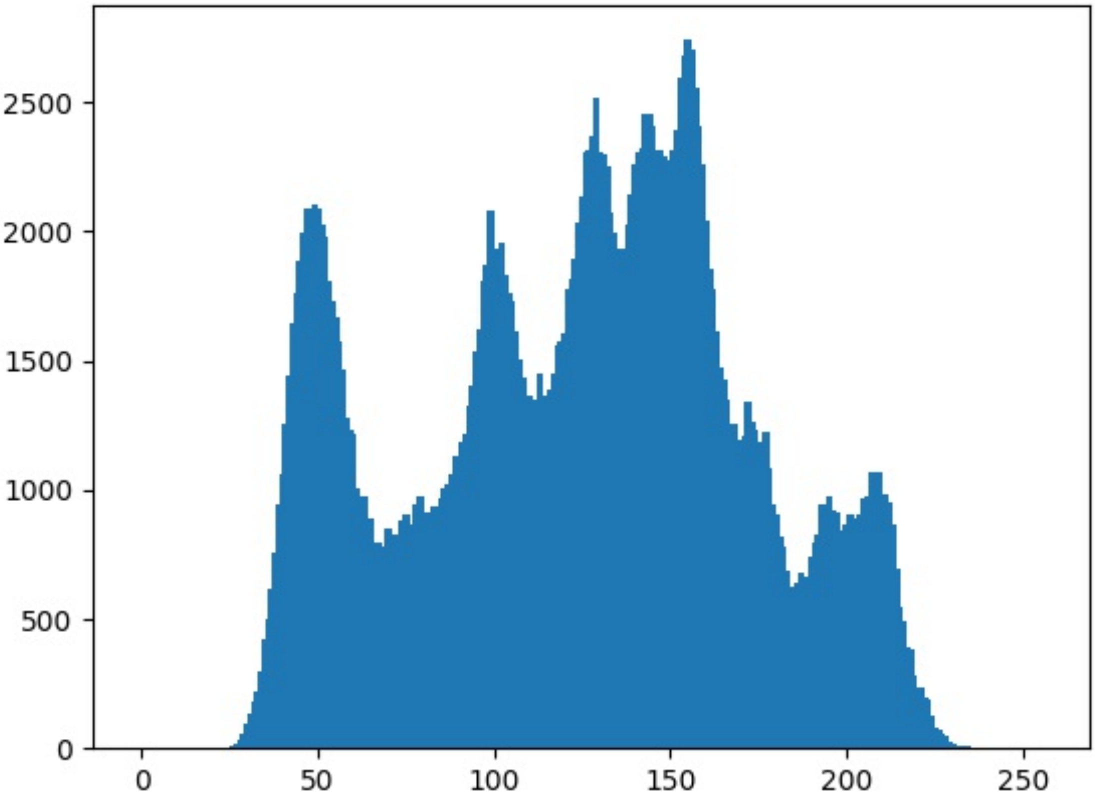
(a)binary image with threshold 128

```
for row in range(height):
        for col in range(width):
                pix = img[row][col]
                if pix >= 128:
                        img[row][col] = 255
                else:
                        img[row][col] = 0
#去掃描每一個像素，當值大於threshold的時候，就把它改成255
```

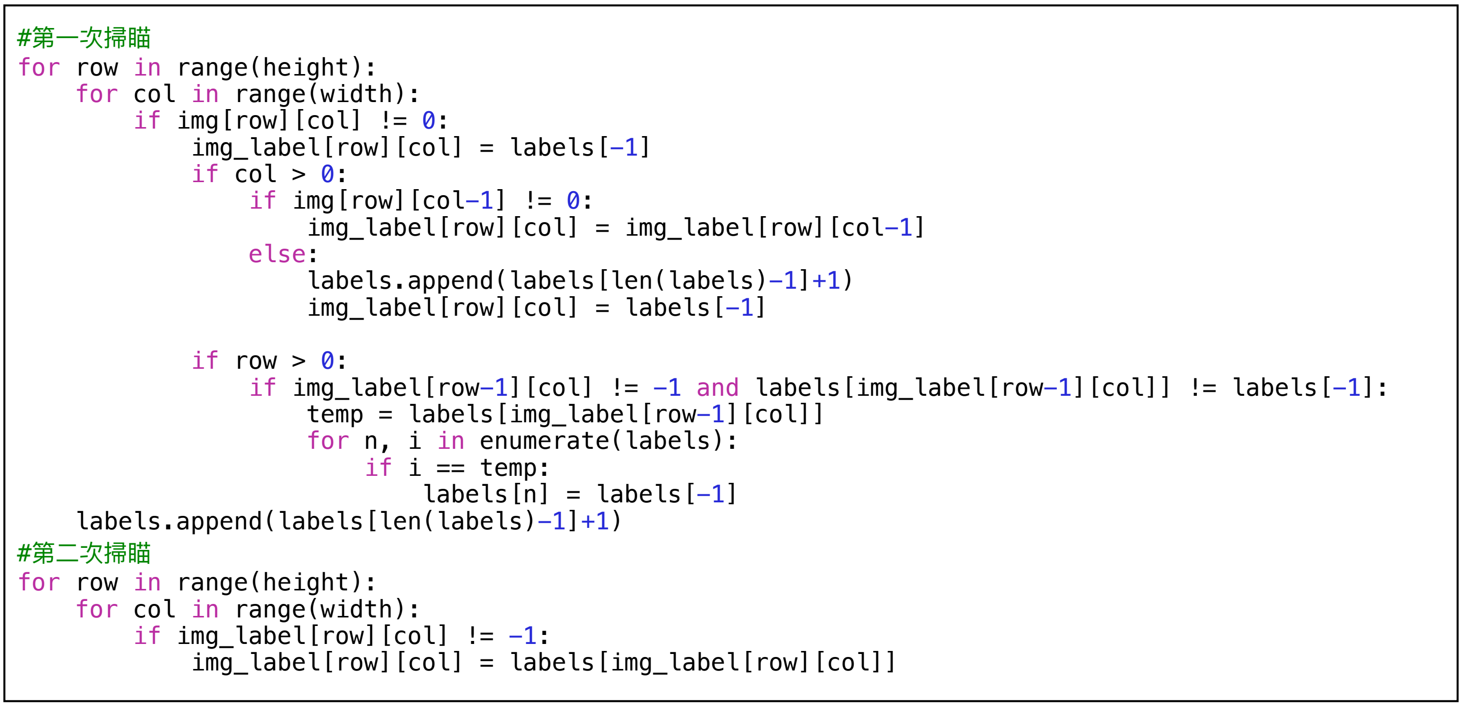


(b)a histogram of lena.bmp

```
for row in range(img.shape[0]):
        for col in range(img.shape[1]):
                pix = img[row][col]
                hist_img[pix] += 1
#掃描每一個像素，並把它統計在一個長度256的list中
```



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



```python
#第一次掃瞄
for row in range(height):
    for col in range(width):
        if img[row][col] != 0:
            img_label[row][col] = labels[-1]
            if col > 0:
                if img[row][col-1] != 0:
                    img_label[row][col] = img_label[row][col-1]
                else:
                    labels.append(labels[len(labels)-1]+1)
                    img_label[row][col] = labels[-1]

            if row > 0:
                if img_label[row-1][col] != -1 and labels[img_label[row-1][col]] != labels[-1]:
                    temp = labels[img_label[row-1][col]]
                    for n, i in enumerate(labels):
                        if i == temp:
                            labels[n] = labels[-1]
    labels.append(labels[len(labels)-1]+1)
#第二次掃瞄
for row in range(height):
    for col in range(width):
        if img_label[row][col] != -1:
            img_label[row][col] = labels[img_label[row][col]]
```

說明：

#img_label:shape與img相同的2D.list，目的是要存放每個像素對應到label編號

#labels: ex.[0, 1, 2, 3, 4, 5, 6, ...]
ex.label4, label5連通=>[0, 1, 2, 3, 5, 5, 6]

#labels主要就是拿來存放標籤的變化結果，位置代表原本的標籤號碼
位置對應的值代表後來經過聯通後的標籤號碼

演算法步驟：（從左上至右下掃描每一個pixel）

1.當pixel的值！＝0，則

2-a.去判斷左邊的pixel是否!=0，若!=0，則
把目前pixel的img_label存成左邊的pixel的標籤號碼

2-b.去判斷左邊的pixel是否!=0，若==0，則
把目前pixel的img_label存成新的一個標籤號碼

3-a.去判斷上方的pixel是否!=0，若!=0，則
把labels中所有的element的值跟上方pixel的img_label相同的element

改成目前pixel的img_label值

4.第二次掃瞄，由img_label與labels

img_label[i] = labels[img_label[i]]