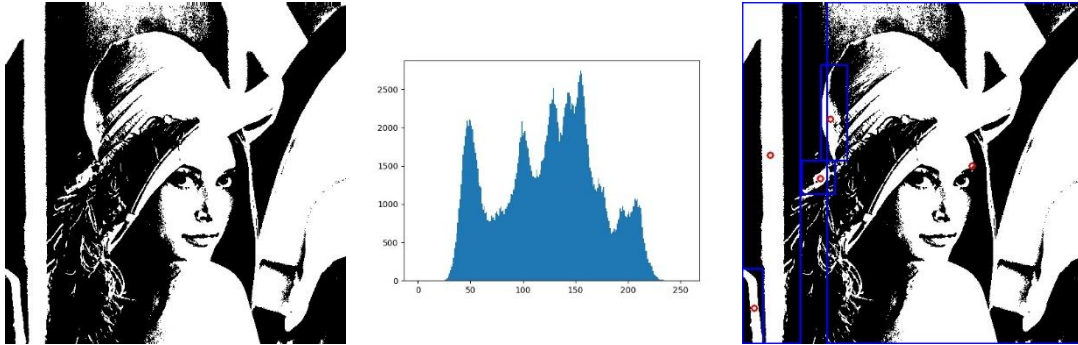


# 2021 CV HW2 Report

## Results



a.

b.

c.

### (a) a binary image (threshold at 128)

兩層 for loop 查看各個 pixel value，若值  $\geq 128$  則設為 255；若值  $< 128$  則設為 0。

```
for i in range(h):
    for j in range(w):
        if img[i][j] >= 128:
            output[i][j] = 255
        else:
            output[i][j] = 0
```

### (b) a histogram

建立兩個 list，一個為直方圖的 X 軸（表示 pixel value 0~255），一個為直方圖的 Y 軸（各 pixel value 的總數）。

兩層 for loop 查看 pixel value 為多少，在相對應的 list 位置將 count 加 1。

```
histogramX = [i for i in range(256)]
histogramY = [0 for _ in range(256)]
for i in range(h):
    for j in range(w):
        histogramY[img[i][j]] += 1
plt.bar(histogramX, histogramY, width=1)
plt.show()
```

## (c) connected components

Step 1. 使用 4 連通的方式，做兩次上到下左到右的 scan。建立一個 list 紀錄各 label 的 root label。Scan 時去查看各 pixel 上方和左邊的 label 值，當兩個不同的 label 相連時，將兩者 root 較小的作為兩者的 root label。

```
def find_root(index):
    root = index
    while flagList[root] != root:
        root = flagList[root]
    return root

def union(index1, index2):
    root1 = find_root(index1)
    root2 = find_root(index2)
    if root1 > root2:
        flagList[root1] = root2
    elif root1 < root2:
        flagList[root2] = root1
# first and second scan
# 略..
```

Step 2. 忽略 pixel count 小於 500 的 label。

```
# omit regions that have a pixel count less than 500
unique, counts = np.unique(label, return_counts=True)
labelCounts = dict(zip(unique, counts))
for key, value in list(labelCounts.items()):
    if value < 500 or key == 0:
        del labelCounts[key]
```

Step 3. 找出各個 label 的所有 row 和 column index，接著找出最大值、最小值畫 bounding box；並計算 centroid 的 x、y，畫出 centroid。

```
# draw bounding box and regions with + at centroid
output = cv2.cvtColor(img, cv2.COLOR_GRAY2BGR)
for key in labelCounts.keys():
    indexX = [i for i in range(len(label[0])) for j in range(len(label))]
    if label[j][i] == key:
        indexY = [i for i, value in enumerate(label) for j in value if j == key]
```

```
x1, y1, x2, y2 = min(indexX), min(indexY), max(indexX), max(indexY)

centroidX = round(sum(indexX)/len(indexX))
centroidY = round(sum(indexY)/len(indexY))
cv2.rectangle(output, (x1, y1), (x2, y2), (255, 0, 0), 2)
cv2.circle(output, (centroidX, centroidY), 4, (0, 0, 255))
```

Note. 將 pixel count 大於 500 的 label region 用隨機的顏色表示出來。由下圖可見，總共有 5 個 region。

