電腦視覺 作業二

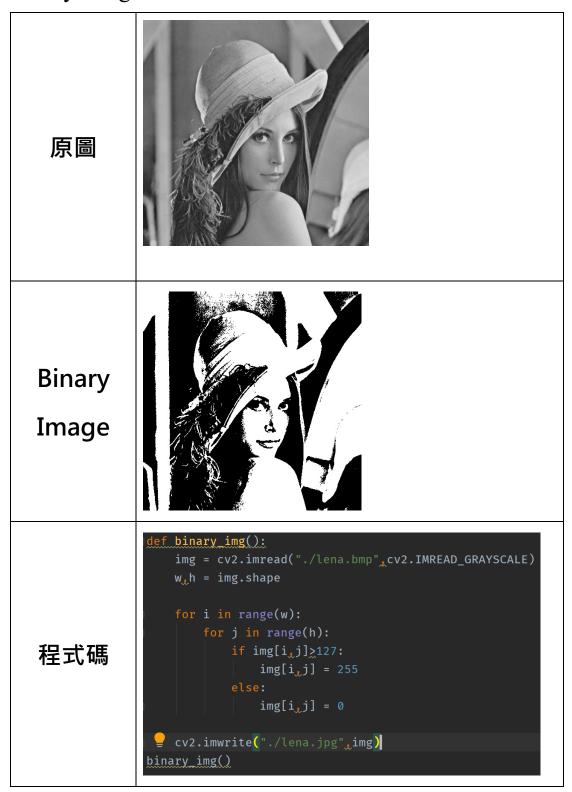
指導老師 傅楸善

學生 蔡宇晴

學號 r08945050

Part1.

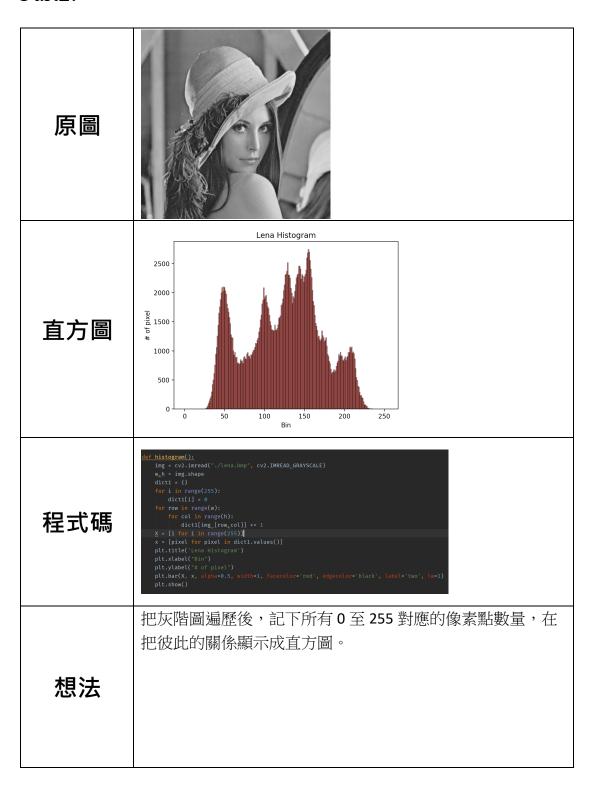
Binary Image



想法

雙 for 迴圈,遍歷所有點後,把閥值上下的點,像素值各自 分成 255 和 0。

Part2.



Part3.



程式碼

```
# print(f"top down, iter = {iter}")
for i in range(h):
    cut = img_counter_pad[i:i+3, j:j+3]
    test = cut * kernel_down

    non_zero_pos = np.where(test != 0)
    non_zero_val = test[non_zero_pos]

for y, x in zip(non_zero_pos[0], non_zero_pos[1]):
    img_counter_pad[i+y, j+x] = np.min(non_zero_val)

# 下往上
print(f"bottom up, iter = {iter}")
for i in range(h-1, -1, -1):
    cut = img_counter_pad[i:i + 3, j:j + 3]
    test = cut * kernel_up

    non_zero_pos = np.where(test != 0)
    non_zero_val = test[non_zero_pos[0], non_zero_pos[1]):
    img_counter_pad[i + y, j + x] = np.min(non_zero_val)
```

```
index500 = []
# 將 500 以下的填 0

9 for c in np.unique(img_counter_pad):
        if np.sum(img_counter_pad == c) < 500:
            img_counter_pad[img_counter_pad == c] = 0

elif c != 0:
        # print(f"c = {c}, # = {np.sum(img_counter_pad == c)}")
        index500.append(c)
```

```
def mark(img, c):
    x_min = np.min(np.where(img == c)[1]) - 1
    x_max = np.max(np.where(img == c)[1]) - 1
    y_min = np.min(np.where(img == c)[0]) - 1
    y_max = np.max(np.where(img == c)[0]) - 1
    center = (np.mean(np.where(img == c)[1]) - 1, np.mean(np.where(img == c)[0]) - 1)
    return (x_min, y_min), (x_max, y_max), center

def draw_cross(img, center, length = 5):
    x, y = center
    x = int(x)
    y = int(y)
    img[y-1:y+1, x - length: x + length + 1] = np.array([0, 0, 255])
    img[y - length: y + length + 1, x-1:x+1] = np.array([0, 0, 255])
    return img
```

```
for i in range(1, len(index500) + 1):
    min_loc, max_loc, center = mark(img_counter_pad, i)
    # print(min_loc, max_loc, center)
    cv2.rectangle(img, min_loc, max_loc, (255, 0, 0), 2)
    img = draw_cross(img, center)
```

首先,先開 2 個 filter,以便分別確認左邊和上方以及右邊和下方的像素點。

步驟 1.

遍歷所有點的時候,在所在位置(I,j)有值的時候,同時確認 左邊及上面,當三者皆有值的時候,所在位置(I,j)用 counter 標記。

步驟 2.

想法

再一次確認左邊及上方,把所在位置(I,j)和其他兩點比較,取出最小值,並賦值給(I,j)。

接著,由右下至左上的遍歷開始執行,此時確認右邊及下方還有(I,j)本身的值。將最小值賦值給(I,j)。以上流程重複 5次,直到不再變化。

步驟 3.

從各自物件中的像素點中,得到所有像素點的座標和座標極大值,座標極小值,進而可以得到重心還有矩形的範圍。