

Computer Vision I _2018

Homework assignment #2

R07522717 機械所製造組碩一 林溫雅

Part1

Description:

Write a program to generate a binary image (threshold at 128, (0-127,128-255))

Algorithm:

使用 PIL 套件之 IO function 讀取原始圖檔，紀錄其中每個 pixel 之灰階值，接著將灰階值大於 127 之 pixels 設為 255，其餘為 0。

Parameters:

im	#讀取原始圖檔
original	#儲存原始圖檔 pixels 之 value
i, j	#迴圈內計數用參數
im_binary	#二元影像

Principal code fragment:

```
#讀取原始資料到一個 2D-array
for i in range (512):
    for j in range (512):
        original[i, j] = im.getpixel((i,j))
        count[original[i, j]] += 1
im_binary = Image.new("L", (512,512), 0)
for i in range (512):
    for j in range (512):
        if original[i,j] > 127:
            #binarize lena.bmp
            im_binary.putpixel((i,j), 255)
#儲存 binarized lene.bmp
im_binary.save("im_binary.bmp")
```

Computer Vision I _2018 Homework assignment #2

Resulting images

binarized lena 二元影像



Part2

Description:

Write a program to generate a histogram.

Algorithm:

在 Part1 讀取原始資料時，已經計算好每個灰階值的 pixels count，在這邊使用這些紀錄好的 count 作圖，使用 matplotlib 輔助製圖。

Parameters:

count	#計算每個灰階值的 pixels count
x, y	#作圖用的 x,y 軸
ax	#作圖用的 image object

Principal code fragment:

#作 histogram 的圖

```
x = np.linspace(0,255,256, endpoint = True, dtype=np.int)
y = count[x]
ax = plt.figure(figsize=(8,4))
ax.set_facecolor((1, 0.8, 0.4))
plt.axes(facecolor='k')
plt.hist(original.flatten(), bins = x, density=False, color='w')
plt.xlabel("gray level value")
plt.ylabel("count")
plt.title("Histogram of lena.bmp")
plt.xlim(0, 255)
plt.ylim(0, max(count))
```

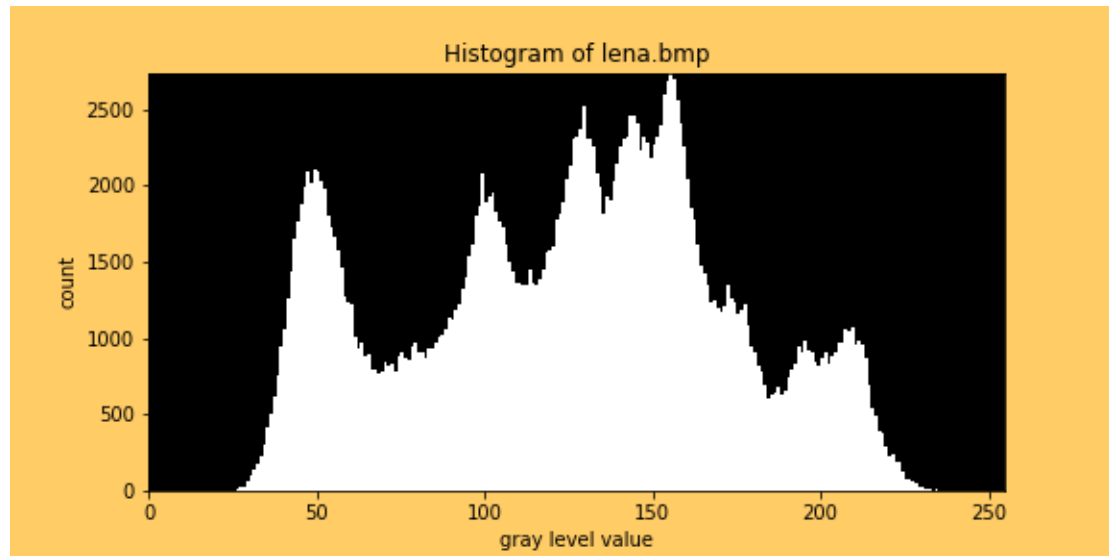
#儲存 histogram 的圖

```
ax.savefig("histogram.png",facecolor=ax.get_facecolor(),
edgecolor='none')
```

Computer Vision I _2018 Homework assignment #2

Resulting images:

Histogram of binarized lena



Part3

Description:

Write a program to generate a connected-components image (regions with bounding box).

Algorithm:

先把每個 pixel 全部標上不同的數字，然後使用 iteration 之方式，重複 top-down 與 bottom-up 直到 pixels 的 label 不再變化為止。

接著計算有哪些是>500 個 pixels 的群體，找出 pixel 數大於 500 之區塊並繪製矩形。

Parameters:

binary	#label pixel 用的矩陣
label1	#label 時使用的參數
anythingchange	#用來判斷 label 還有沒有在變化
loops	#計算 loop 次數
pixelcount	#計算有哪些是>500 個 pixels 的群體
area	#儲存>500 的區域資訊
connected_component	#connected component 影像物件
draw	#預備繪製矩形之參數
minx, miny, maxx, maxy	#計算矩形對角點用

Principal code fragment:

```
#先把每個 pixel 全部標上不同的數字
for i in range (512):
    for j in range (512):
        if binary[i,j] != 0:
            label1 += 1
            #此時 label1 是 total 非 0 之 pixels 數
            binary[i,j] = label1

#使用 iteration 之方式
while (anythingchange == True):
```

Computer Vision I _2018 Homework assignment #2

```
#接著 topdown
    #先左到右
    anythingchange = False
    for i in range (0,512):
        for j in range (1,512):
            if binary[i,j] * binary[i,j-1] != 0 and binary[i,j-1] <
binary[i,j]:
                binary[i,j] = binary[i,j-1]
                anythingchange = True
        #再上到下
        for j in range (0,512):
            for i in range (1,512):
                if binary[i,j] * binary[i-1,j] != 0 and binary[i-1,j] <
binary[i,j]:
                    binary[i,j] = binary[i-1,j]
                    anythingchange = True
#接著 bottomup
    #先右到左
    for i in range (0,512):
        for j in range (510, -1, -1):
            if binary[i,j] * binary[i,j+1] != 0 and binary[i,j+1] <
binary[i,j]:
                binary[i,j] = binary[i,j+1]
                anythingchange = True
        #再下到上
        for j in range (0,512):
            for i in range (510, -1, -1):
                if binary[i,j] * binary[i+1,j] != 0 and binary[i+1,j] <
binary[i,j]:
                    binary[i,j] = binary[i+1,j]
                    anythingchange = True
    loops += 1
```

Computer Vision I _2018 Homework assignment #2

Resulting images:

Connected component with rectangle boundaries

