Computer Vision I _2018

Homework assignment #2

R07522717 機械所製造組碩一 林温雅

```
#使用 python
#import 套件
%matplotlib inline
import matplotlib.pyplot as plt
from PIL import Image
import numpy as np

#讀取 lena.bmp
im = Image.open("lena.bmp") #讀取原始圖檔

#先儲存原始資訊
original = np.zeros([512,512],np.int ) #儲存原始圖檔 pixels 之 value
binary = np.zeros([512,512],np.int ) #儲存 binary 圖檔 pixels 之 value
count = np.zeros(256, np.int)

#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)
            #預備第三題所需之矩陣
            binary[i,j] = 1
```

```
#儲存 binarized lene.bmp
im_binary.save("im_binary.bmp")

#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')



#下面開始第三題
label1=0
anythingchange = True
loops = 0
#binary 已經是 initialize 過的 512*512 matrix 了

#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):
    #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

#計算有哪些是>500 個 pixels 的群體
pixelcount = np.zeros(133960, np.int)
```

```
area = np.zeros(0, np.int)

for i in range (512):
    for j in range (512):
        if binary[i,j] != 0:
            pixelcount[binary[i,j]] += 1

#找出 pixel 數大於 500 之區塊
for i in range (133960):
    if pixelcount[i] > 500:
        area = np.append(area, i)

#繪製矩形
from PIL import ImageDraw
boundary = np.zeros([np.size(area), 4],np.int )
connected_component = im_binary.convert('RGB')
draw = ImageDraw.Draw(connected_component)
for k in range(np.size(area)):
    minx, miny, maxx, maxy = 600, 600, 0, 0
    for i in range (512):
        for j in range (512):
            if binary[i,j] == area[k]:
                if i < minx:
                    minx = i
                elif i > maxx:
                    maxx = i
                if j < miny:
                    miny = j
                elif j > maxy:
                    maxy = j
    draw.rectangle([minx, miny, maxx, maxy],fill=None, outline="red")

#儲存 connected_component 之圖片
connected_component.save("connected_component.bmp")
```