Computer Vision I \_2018

Homework assignment #2

R07522717機械所製造組碩一 林温雅

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

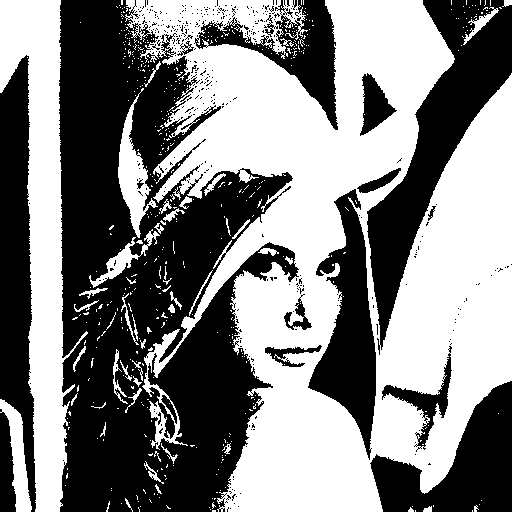
im\_binary.putpixel((i,j), 255)

#儲存binarized lene.bmp

im\_binary.save("im\_binary.bmp")

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)

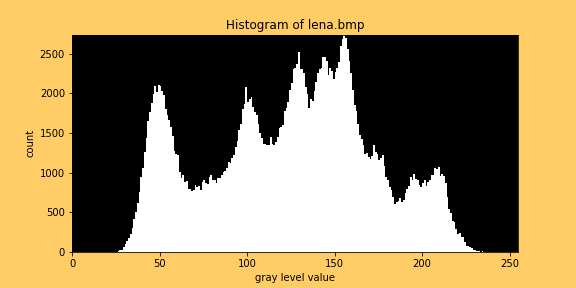
plt.ylim(0, max(count))

#儲存histogram的圖

ax.savefig("histogram.png",facecolor=ax.get\_facecolor(), edgecolor='none')

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):

#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

Resulting images:

Connected component with rectangle boundaries

