

高等電腦視覺

作業#(02)

姓名： 吳佩霖

學號： 111c52016

指導老師： 張陽郎

程式執行說明

1.

首先選擇要執行 C++ 版本或 openCV 版本

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version: _
```

2.

本次無分題號直接將個別版本所有結果執行出來

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:2
CC_STAT_AREA:186832
CC_STAT_AREA:142462
CC_STAT_AREA:262258
```



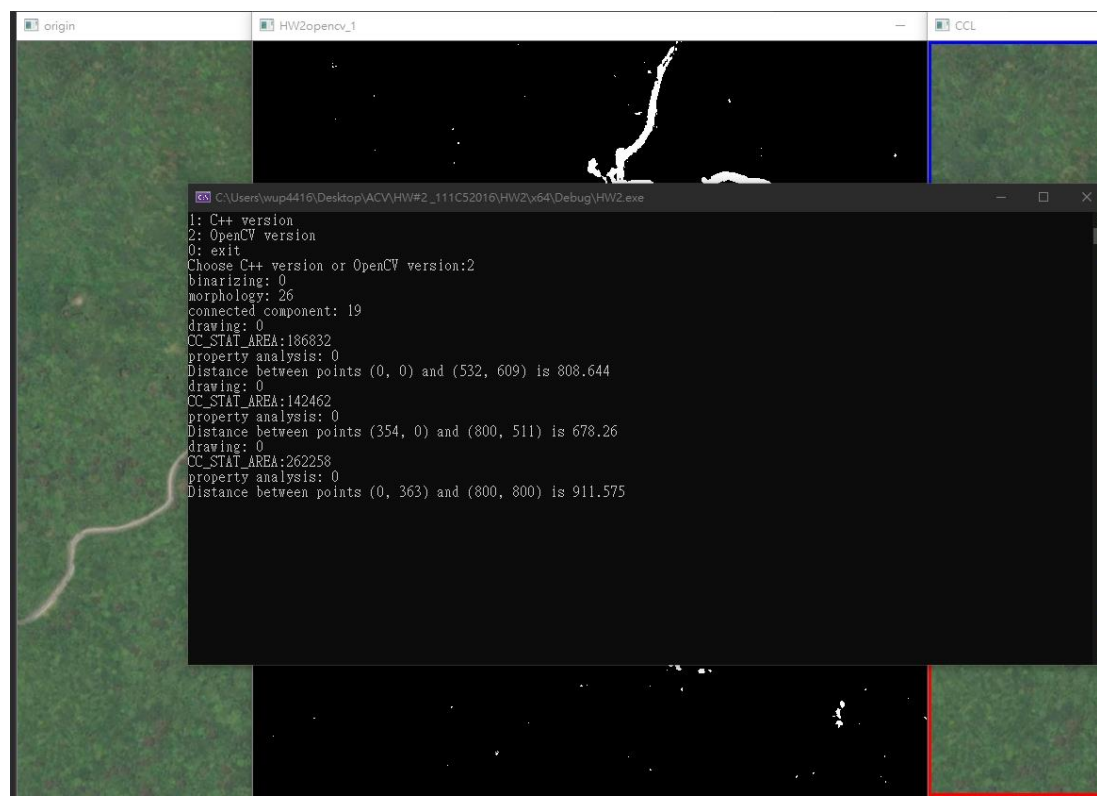
3.

執行完成後會看到”file has been saved 檔名.bmp successful!”的提示，檔案則儲存於 HW2 的資料夾中。

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:1
file has been saved as HW2_C++_1.bmp successful
file has been saved as HW2_C++_2.bmp successful
file has been saved as HW2_C++_3.bmp successful
=====
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version: _
```

4.

若是執行 openCV 的版本則是會跳出結果的視窗，並如同 C++版本一樣有”file has been saved 檔名.bmp successful!”的提示，即為儲存完成。



1.

C++

HW2_C++_1.bmp



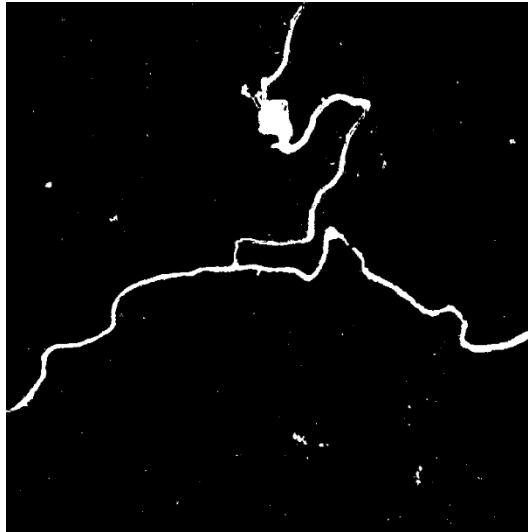
Discussion_1

將原圖讀入，進行二值化，這邊是取 RGB 三通道的值加起來/3 小於 137 來進行，出來成果雖然雜訊不多，但道路多處斷裂，需要進行膨脹來填回。

1.

OpenCV

HW2opencv_1.bmp



Discussion_1

將原圖使用 `imread` 讀入，在這邊為了減少雜訊，先使用了 `blur` 將圖片進行模糊處理，之後使用 `threshold` 進行二值化，利用 `THRESH_OTSU` 進行處理，最終得出上圖結果。

2.

C++

HW2_C++_2.bmp



Discussion_2

將二值化後的結果進行膨脹處理，雖然部分斷裂的路徑獲得修復，但相對的雜訊也跟著放大了許多。

2.

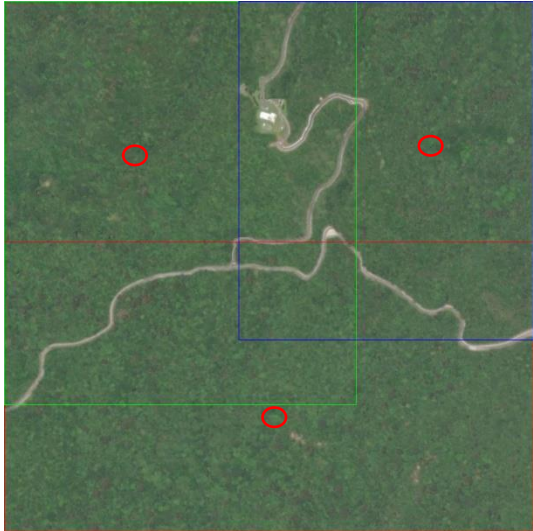
OpenCV

HW2opencv_2.bmp



Discussion_2

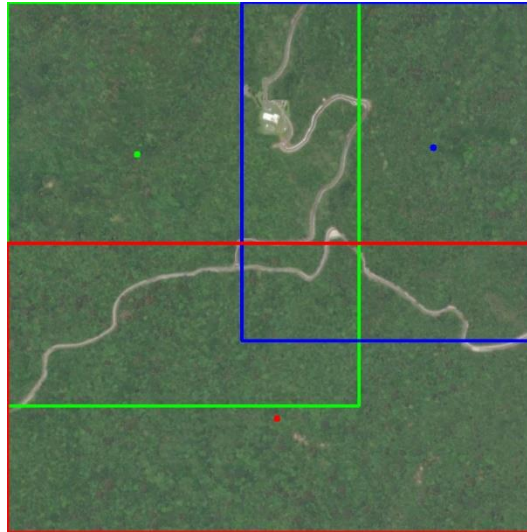
經過模糊與二值化的圖片已經將道路輪廓明顯描繪出，但因為有部分雜訊與道路斷裂的問題，在這邊使用形態學 MORPH_OPEN(先侵蝕後膨脹)來去除散落雜點，之後使用 MORPH_DILATE(膨脹)將路連貫起來，雖然還是有多個大型雜訊，但比起一開始只進行二值化的圖片已經好很多，且道路都為連貫無斷裂。

3.
C++
HW2_C++_3.bmp

property analysis
<pre> C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\64\Debug\HW2.exe 1: C++ version 2: OpenCV version 0: exit Choose C++ version or OpenCV version:1 binarizing: 6 file has been saved as HW2_C++_1.bmp successful morphology: 186 file has been saved as HW2_C++_2.bmp successful connected component: 436 Distance between points (800, 800) and (800, 800) is 0 Distance between points (0, 0) and (799, 436) is 910.218 Distance between points (0, 191) and (531, 990) is 959.355 Distance between points (354, 289) and (1153, 1088) is 1129.96 label route1 : area = 262258 centroid = (407, 171) label route2 : area = 186832 centroid = (196, 569) label route3 : area = 142462 centroid = (643, 579) property analysis: 12 drawing: 18 file has been saved as HW2_C++_3.bmp successful 1: C++ version 2: OpenCV version 0: exit Choose C++ version or OpenCV version: </pre>
Discussion_3
<p>C++版本的 bounding box 跟圓心因為只畫一個 pixel，無調整過粗細，因此在此圈起中心點的結果表示。connectedComponent 的判斷方式為如果 pixel 值為 0 則做標記，以標記後的結果框出個別區域。</p> <p>計算利用計算出區域再計算出中心點並且直接給值，而在執行結果也都附上所有數據。</p>

3.

OpenCV

HW2opencv_3.bmp



property analysis

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:2
CC_STAT_AREA:186832
CC_STAT_AREA:142462
CC_STAT_AREA:262258
-
```

C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe

```
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:2
binarizing: 1
morphology: 25
connected component: 22
drawing: 0
CC_STAT_AREA:186832
property analysis: 1
Distance between points (0, 0) and (532, 609) is 808.644
drawing: 0
CC_STAT_AREA:142462
property analysis: 0
Distance between points (354, 0) and (800, 511) is 678.26
drawing: 0
CC_STAT_AREA:262258
property analysis: 0
Distance between points (0, 363) and (800, 800) is 911.575
```

```
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:
```

Discussion _3

最後將進行完形態學的圖片利用 connectedComponents 標籤出三大區域，並且計算出中心畫上中心點，計算出每一區塊面積。但由於經過膨脹侵蝕處理過後的圖片路徑已經有些許變形，框出來的成果回到原圖上並不精確，但大致上的方向都沒太大問題。

4.

執行時間比較

Process	C++	OpenCV
binarizing	6 ms	1 ms
morphology	248 ms	26 ms
connected component	436 ms	20 ms
property analysis	11 ms	0 ms
drawing	17 ms	0 ms

C++

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:1
binarizing: 6
file has been saved as HW2_C++_1.bmp successful
morphology: 248
file has been saved as HW2_C++_2.bmp successful
connected component: 436
Distance between points (800, 800) and (800, 800) is 0
Distance between points (0, 0) and (799, 436) is 910.218
Distance between points (0, 191) and (531, 990) is 959.355
Distance between points (354, 289) and (1153, 1088) is 1129.96
label route1 : area = 262258 centroid = ( 407, 171)
label route2 : area = 186832 centroid = ( 196, 569)
label route3 : area = 142462 centroid = ( 643, 579)
property analysis: 11
drawing: 17
file has been saved as HW2_C++_3.bmp successful

1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:
```

OpenCV

```
C:\Users\wup4416\Desktop\ACV\HW#2_111C52016\HW2\x64\Debug\HW2.exe
1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:2
binarizing: 1
morphology: 26
connected component: 20
CC_STAT_AREA:186832
Distance between points (0, 0) and (532, 609) is 808.644
CC_STAT_AREA:142462
Distance between points (354, 0) and (800, 511) is 678.26
CC_STAT_AREA:262258
Distance between points (0, 363) and (800, 800) is 911.575
property analysis: 0
drawing: 0

1: C++ version
2: OpenCV version
0: exit
Choose C++ version or OpenCV version:
```

Discussion _4

以上方數據來看，明顯在 OpenCV 的時間遠優於 C++的版本，以上方成果看來，OpenCV 在進行二質化與形態學所呈現的成果也比 C++好的多。

connected component 的部分因為 C++版本需要將所有像素逐一便臨再做出判斷，因此使用時長也相對久的多。

在分析圖片的部分，因為 C++在這邊沒做成長軸計算，因此在時間處理上跟 opencv 版本的差異參考價值有待商榷。