

數位視覺效果—Project 2

組員: B06902062 資工三 陳法熏

B06902112 資工三 邱睿成

Instruction:

我們的程式因為執行時間都滿花時間的，為了減少測驗時間，因此會先將照片 scale 0.1 倍。

`feature.py`:

讀入所有的照片，最後跑出來的結果是倆倆照片的 pair，總共有 5 組 pair。

`final.py`:

因為有 import `feature.py`，直接執行 `final.py` 即可得到 scale 0.1 倍後的全景照片了

因為上傳的 code 是跑 scale 0.1 倍的照片，因此有個 6000*4000 的檔案夾是

我們自行測試跑原圖所出來的結果（從右到左拼上照片的過程）

分工：

B06902062 資工三 陳法熏： 找出 feature、投影、求 f 值

B06902112 資工三 邱睿成： matching、拼接、拍照、統整

拍攝地點：

社圖

拍攝照片：

6 張

拍攝照片時注意事項：

必須架腳架，且確認解架上的水平儀讓相機能水平拍攝，拍照時固定相機的 f 。

Feature:

在找 feature 點的部分，原本是選擇做 Multi-Scale Harris corner detector，由於印象中老師說每一層的結構基本上跟 Harris corner detector 一樣，於是就先寫一層的再做多層，殊不知其實細部有一些小差異，例如 corner response function。

在計算 gradient 時遇到一些麻煩，原本用上課所說的 $[1, -1]$ 的 filter 效果不佳，後來改用 opencv 裡的 sobel 函式得到更好的效果；在計算 corner response 時使用 numpy 的 `LA.eigvals()` 來計算 λ_1, λ_2 。在實作 Non-maximal suppression 時沒有遇到太多困難，就如同上課所說的先 sort 完再依半徑大小一個一個放入夠遠的最大值，選用圖行的寬作為初始的半徑。

Feature 效果：



Matching:

在一開始進行 matching 時，以為 MSOP 做出來的效果會比較好，因此花了許多時間寫 MSOP，但後來發現在有使用腳架的情況，且相機是完全水平且沒有仰角的狀況下旋轉拍攝，照片是不會有旋轉的問題，因此不需要用到 MSOP，所以最後採取的方法為最簡單的方式：

對所有的特徵點，取周圍 5×5 的 pixel 之能量，同除以 255，並存入此特徵點的大小為 25 之向量裡。將相鄰照片所有的特徵點拿來做比較求出能對起來的特徵點之在兩個圖上的座標 pair : (Lx, Ly, Rx, Ry)

Pair 求法:

以左邊圖的特徵點為基準 L，讓每個特徵點都去爆搜一次右邊圖的所有特徵點，找出特徵點向量距離的最小值 Lmin 與第二小值 Lsecmin，找到 Lmin 後，以這個右邊圖之特徵點為新的基準 R，爆搜一次左邊圖的所有特徵點，也是找出最小值 Rmin 與第二小值 Rsecmin，看 Rmin 是不是特徵點 L，如果是互相為對方的最小值我才會選擇。在 L 選到互為最小值的 R 後，我設了第一個 threshold，在滿足 $Lmin \leq (Lsecmin \times 0.4)$ 且 $Rmin \leq (Rsecmin \times 0.4)$ 的條件下，進入到第二個 threshold，而這個 threshold 條件就比較單純，把 L, R 的 (x, y) 相減得出 dx, dy 後，只有 dx, dy 小於我所設的大小後，(L, R) 最終才會被我選成一對 pair，在找出所有滿足條件的 pair 之後，就完成了 matching，並把 pair 的結果傳給 Stitching。

Matching 效果：

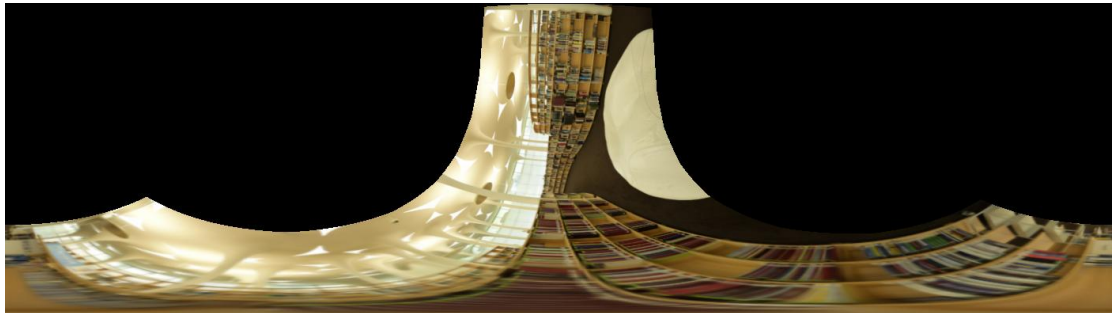
我們總共有 5 張 2 張相鄰照片所找出來的特徵點連線，而以下是做出來的

結果之一



投影：

Projection 的部分比較需要重新思考兩座標之間的關係，上課時給的公式是由平面投影到圓柱的公式，但上課也說過在做 warping 時要使用 inverse warping 的方式才不會出現黑點，於是需要重新理解兩座標之間的關係來算出齊 inverse function。focal length 的取值原本是要參考 autostitch 給的資訊，但做出來的圖片不知為何效果奇差(如附圖)便打消了這個念頭，用手動的方式找出最適合的 focal length



投影效果：



拼接：

拼接的方法很簡單，把照片從右邊往左邊拼，利用各個照片中的 pair 求出將右邊的照片要移動多少 dx dy 才能跟左邊的照片何在一起，合再一起後再進行 blending，得出兩張照片拼接的結果。

而組成大照片的方式也是從右往左拼，先將第 5 張與第 6 張照片和起來成 result5_6，再利用第 4 張與第 5 張的 pair 關係將 reult5_6 拼到第 4 張上，以此類推得到最後的結果。

拼接結果：

拼接兩張：



最終結果：

