

# VFX Project 2 - Image Stitching

b08902109 邱俊茗

b08902106 袁昕德

## focal length estimation

利用auton stitch這個軟體產生的txt檔中可以知道focal length是837左右，但實際運行時發現837太小，而txt檔中的矩陣看起來像是讓圖縮小了4倍，所以推測837\*4才是正確的focal length，而實際執行時也確實得到了很好的效果。

## feature detection

在本次作業，我們使用的feature detection的演算法是Multi-Scale Oriented Patches [1]。建造pyramid時，paper沒有說明如何決定要建幾個level，因此我們決定level=0是原始解析度(h,w)，level=1是(h/2, w/2)，level=2是(h/4, w/4).....，直到寬或高小於60 pixel時就不再繼續砍半。考慮到我們使用的圖片還蠻大的(4032x3024)，如果每個level都挑選feature的話，level=0的feature會算挺久的，而且就算從長寬少一半(2016x1512)的level=1開始找，都還能挑到幾千個feature，因此我們的程式有一個參數是base\_level，代表建完pyramid後要挑第幾個level以後的圖片擷取feature，預設是1代表原始解析度的level=0不會進行feature detection（但是最後stitching的時候還是用原始解析度的圖片進行）。

參考paper的作法及參數，對每一個level的image，算出每個pixel的Harris matrix後，我們使用了[1]這篇paper使用的harmonic mean作為interest point detection function。

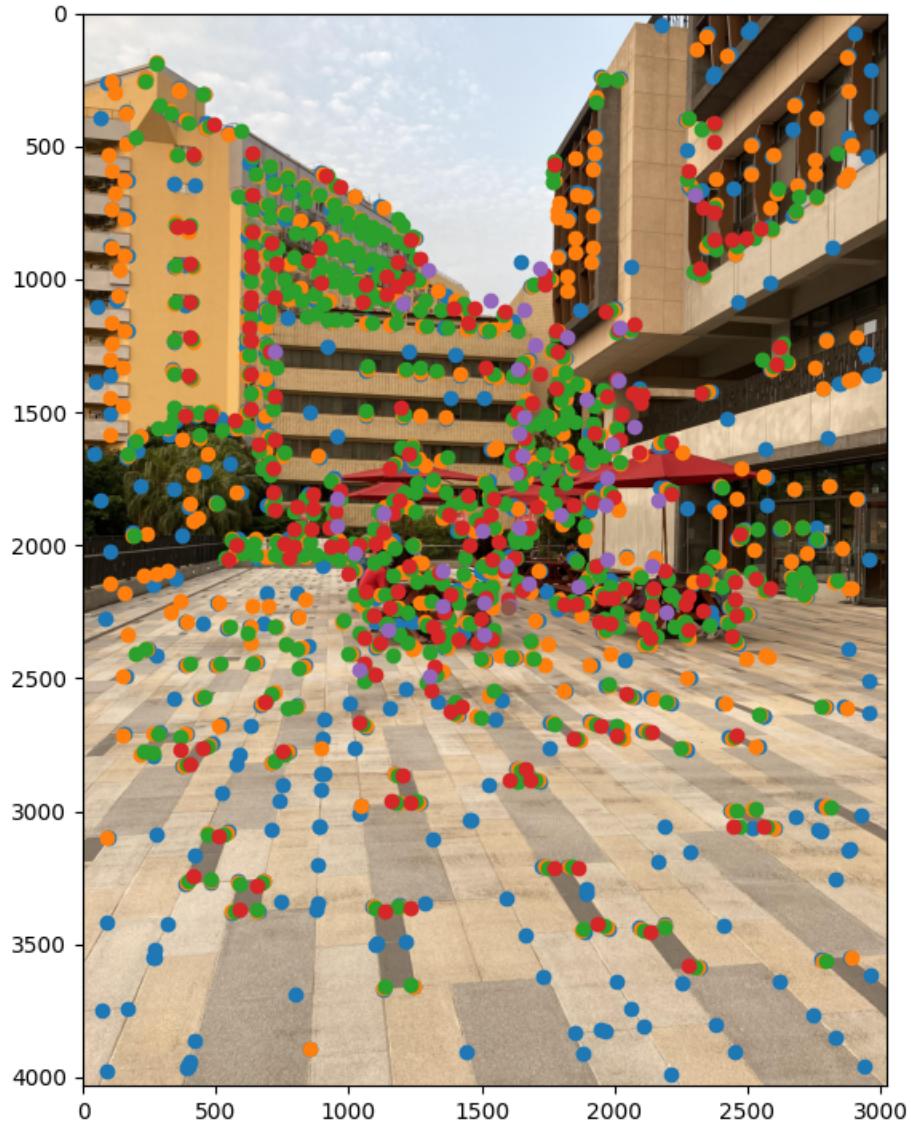
算出每個pixel的interest point detection function  $f(x, y)$ 後，對每個(x, y)，如果  $f(x, y)$ 是周圍3x3正方形裡的極大值而且  $f(x, y) > 10$ 的話，就進行Sub-pixel Accuracy的調整，得到 $(x', y')$ 跟 $f(x', y')$ ，然後把 $(x', y', f(x', y'))$ 插入一個candidate list裡面，但是有時候 $(x', y')$ 會跟 $(x, y)$ 差很遠，發生時通常代表 $f(x, y)$ 跟周圍3x3正方形裡的第二大的值沒有差很多，我們認為這樣的(x, y)通常不會是好的feature，而且這樣的情況也不常發生，因此我們會直接跳過不加入candidate list。

接下來要幫每個candidate找Oriented Patch，如同paper的作法找出major orientation後抓出8x8的patch並normalize 64個點的亮度，得到64維的feature descriptor，但是靠近圖片邊緣的candidate可能patch會取到圖片外的點，這時我們會移除這個candidate。

之後要進行Adaptive Non-maximal Suppression，我們發現paper上的作法跟上課敘述的作法好像有一點不太一樣，因此決定遵照我們對paper所理解的作法。對每個candidate  $(x_i, y_i, f_i)$ ，賦予一個suppression radius：

$$r_i = \min_{j \neq i} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \text{ s.t. } f_i < 0.9f_j$$

然後選suppression radius前500大的作為該level最後留下來的features。



偵測到的feature, 不同顏色代表不同level的feature。可以看到許多磁磚角落跟建築物角落都有被挑選出來。

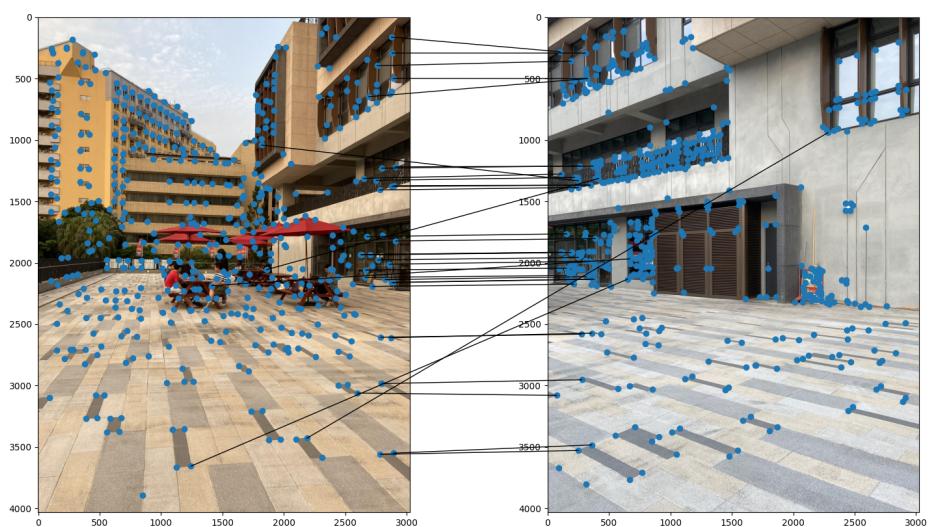
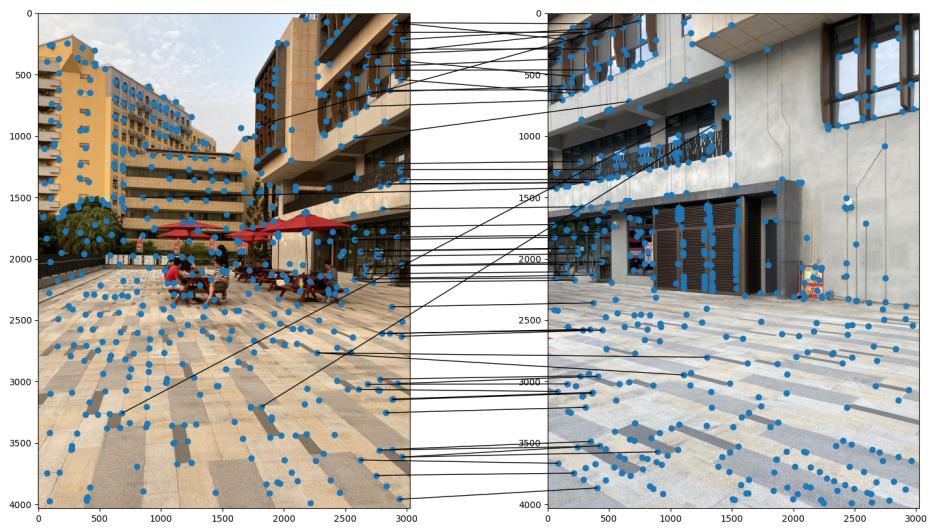
## feature matching & image matching

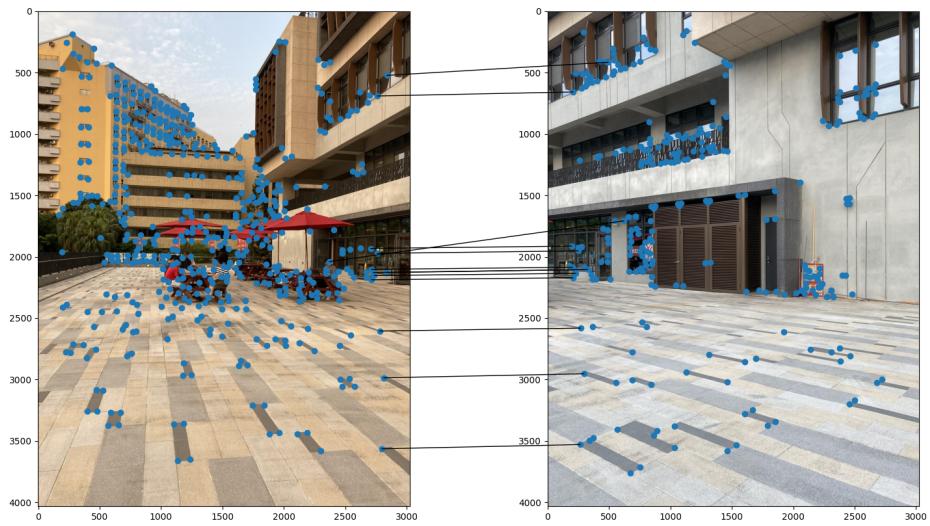
有了兩個圖片的各個level的features(我們假設每一張圖片有同樣的大小, 因此最後取出的level數量一致), 我們可以把feature的座標投影到圓柱座標, 並找出最恰當的平移。

投影到圓柱的方式, 我們採取以圖片的中心為原點, 向右是正x軸, 向上是正y軸的座標, 將圖片投影到半徑為focal length的圓柱上(圓柱與平面切於y軸), 然後推算座標之間的轉換, 得到平面與圓柱之間的兩種轉換函式。

我們的做法無法自動判斷哪兩張圖片是相接的，必須給定一個image list  $I_1, I_2, \dots, I_n$ ，其中  $I_1, I_2$  相接， $I_2, I_3$  相接...， $I_{n-1}, I_n$  相接， $I_n, I_1$  也相接。至於是  $I_1, I_2$  是往右還是往左我們的作法會自動判斷。

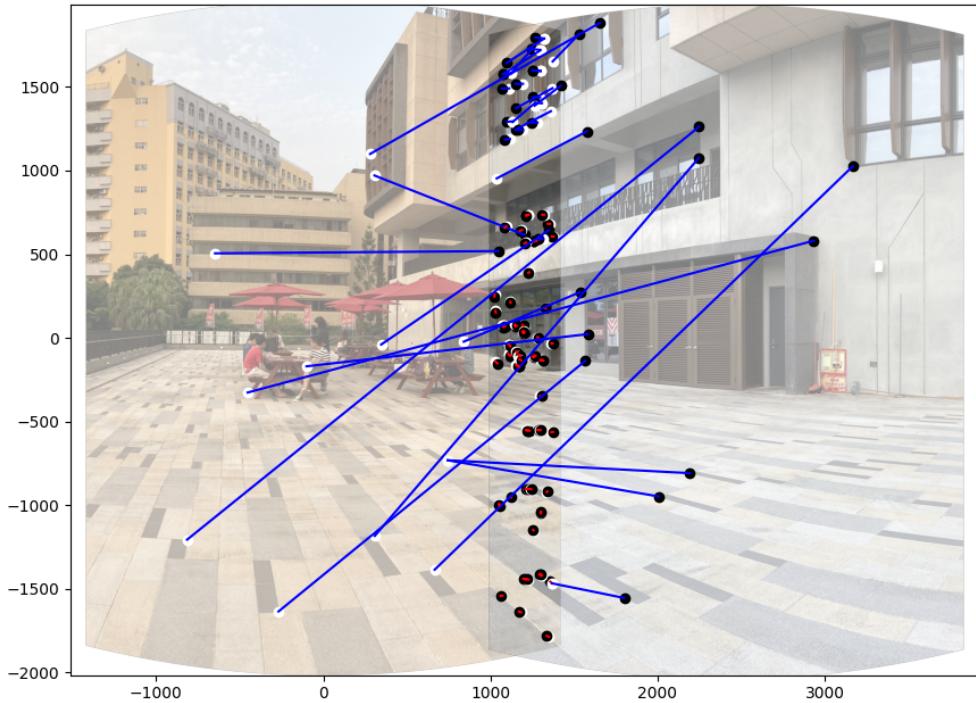
我們還要找兩個相接的圖片同一個level的feature之間的對應關係。一個簡單的作法是對圖片A level l的每個feature，找圖片B level l中 feature descriptor 的 Euclidean distance 最近的k個當作可能的配對。但是這樣會有很多錯誤的配對，會讓之後的RANSAC比較難做。因此需要一個篩的方式。[1]跟[2]提到，如果使用一個閾值d，然後只留下Euclidean distance  $< d$  的配對，這樣的作法通常不會很好。好一點的作法是A的feature  $f_A$  找B的feature中最近的  $f_{B1}$  跟第二近的  $f_{B2}$ ，如果  $|f_A - f_{B1}| < 0.65|f_A - f_{B2}|$ ，則把  $f_A, f_{B1}$  的配對留下來，如此每個  $f_A$  只會找到最多一個配對。考慮到跟  $f_A$  非常接近的  $f_B$  可能有不只一個，我們稍做了修改，找跟  $f_A$  最近的前k名  $f_{B1}, \dots, f_{Bk}$ ，找其中  $f_A$  與後  $k/2$  名的距離的平均當作d，把  $f_A$  與前  $k/2$  名中距離小於  $0.65d$  的配對保留下來（我們使用了  $k=4$ ）。為了對稱性，我們也讓B的feature去找A的feature，並且取兩個pair集合的交集，就會變成剩下的可能配對。





兩張圖片中，其中3個level的feature找出的可能的配對。經過以上的篩選，可以發現只有小部份的feature會順利找到配對，但是剩下來的配對有很大的比例都是正確的。這有利於之後進行的RANSAC。

有了候選的對應關係，我們可以將feature的位置投影到圓柱座標，用RANSAC找出inlier最多的平移。我們挑 $k=3$ 個pair，找出他們的平均位移 $\text{delta}_x$ ,  $\text{delta}_y$ ，並計算所有的pair中有幾個的位移跟 $(\text{delta}_x, \text{delta}_y)$ 誤差小於 $\text{epsilon}$ （我們取 $\text{epsilon}=\min(\text{寬}, \text{高})/200$ ）。跑500次iteration，其中最多人同意的當作最後的位移。



RANSAC的結果，白色的點是左邊的圖的feature位置，黑色的點是右邊的圖的feature位置，紅色的邊是inlier的配對，藍色的邊是outlier的配對。注意到接縫處的上半部份，有很多配對是相近但是接不太起來的，只有平移的話很難讓上面的配對跟下面的配對都很好的對齊，看起來還要有旋轉才能兩邊都符合，我們認為有可能是相機的仰角非零的關係。理論上如果相機的仰角是0(完全水平)，投影到圓柱座標後是只用平移(甚至只有水平方向平移)就拼的起來的。

## blending

利用相鄰兩張圖片之間座標的位移，我們可以得到每張圖片相對於第一張圖在圓柱上的位移，特別地，我們會再原本的最後一張圖片後面再接一張第一張圖片，並為了讓兩張第一張圖片的垂直座標一樣，需要將整個環視圖的每個點減一個垂直的位移，讓最後一張圖跟第一張圖能順利接合。而在獲得最後的環視圖時，我們會將每張圖根據與第一張圖在圓柱上的位移，將投影到圓柱上的圖片複製到環視圖上，並賦予某個權重(我們選擇與左右邊界較近的距離的三次方作為權重)，至於重疊的部分，則利用這個權重進行加權平均。由於位移不一定是整數，因此在複製到環視圖的過程中，我們是將環視圖的格子點(取樣點)進行座標轉換到原本的照片中進行插值，並且因為第一張圖的左邊並沒有跟最後一張圖blending，所以在複製時我們是取第一張的中心到第二個第一張的中心，以確保所有的交界處都有經過blending處理。

# Result



總體來看，大部分的邊界接合的很好，並沒有發生明顯的斷裂，但在一些細節(窗戶等等)可以看出有些鬼影，是由於沒有完全疊合的因素，只是blending淡化了痕跡以至於不明顯，推測可能是因為相機的仰角並不為0，以及相片的橫軸並不平行於地面。

# Reference

- [1] Multi-Scale Oriented Patches (Matthew Brown, Richard Szeliski, and Simon Winder)  
[https://www.researchgate.net/publication/239917360\\_Multi-Scale\\_Oriented\\_Patches](https://www.researchgate.net/publication/239917360_Multi-Scale_Oriented_Patches)
- [2] Distinctive Image Features from Scale-Invariant Keypoints  
<https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf>