

VFX HW2 Paranoma

Report

B03901133 電機三 施順耀

B03901009 電機三 郭笛萱

1. MSOP:

(1) feature detection:

在不同 scale 下，利用 multi scale harris corner detector 找出 feature point。



(2) feature descriptor:

以每個 feature point 為中心，考慮 gradient，展開 40×40 的 window（若在當前維度無法展開則捨棄這個 feature point），接著再以這個 window 每 5×5 取一個平均值，組合成一個 8×8 的 window，最後則以此 feature point 的位置和這 8×8 的 element 總共 66 維的 vector 當做這一個 feature point 的 feature descriptor。

2. feature matching

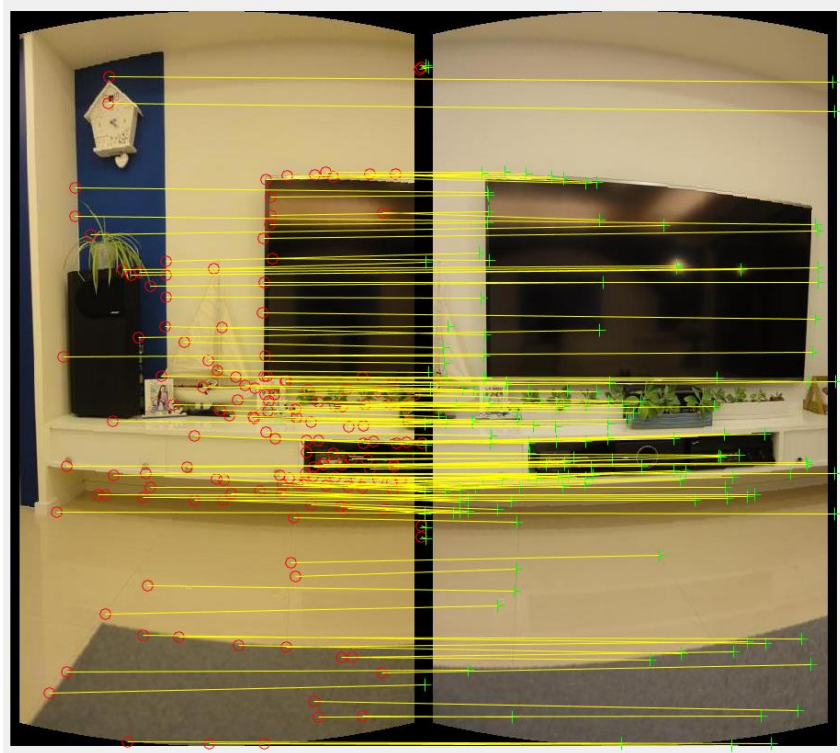
每一個 feature matching 的方法，我們都會做雙向的 match，也就是要兩點互為最佳的 matching point，這兩個 feature point，才會被當作一個 pair。另外我們還加上了 geometric constraint，也就是一個 pair 中的兩個 feature

point 的 y 座標的的差必須小於一個數值(在我們這次的作業中，設定為 10 個 pixel)

尋找 matching pair 的方法我們使用了兩種：

A. exhaustive matching

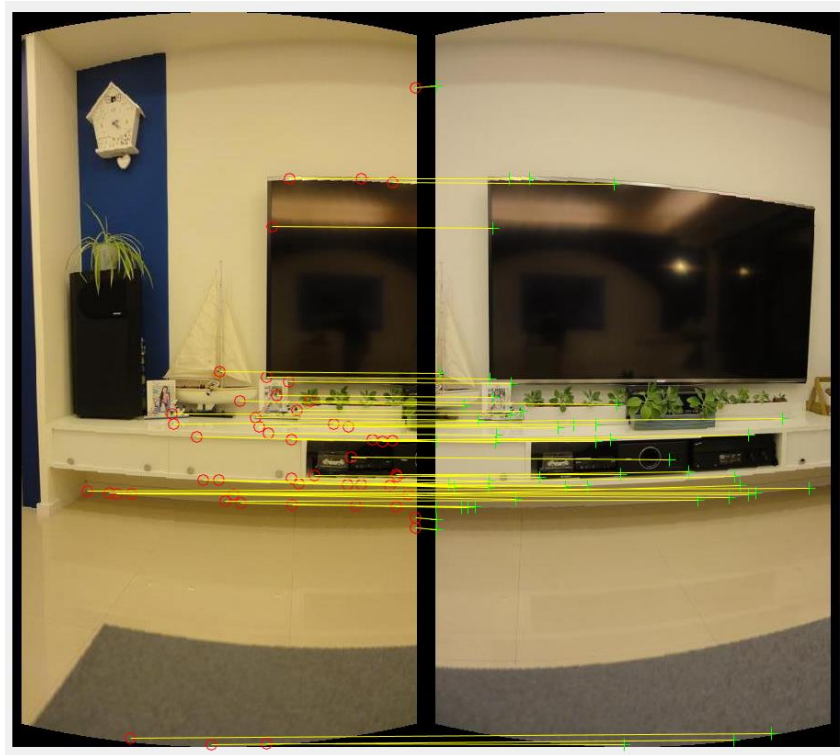
對於一組照片 pair，A 和 B，我們對於 A 裡面的每一個 feature point，去算其對於 B 裡面所有的 feature point 的 euclidean distance，並把所有的 pair 記錄下來。



B. BONUS wavelet-based hash matching:

在 MSOP 最後的 descriptor，我們做 wavelet transform，之後再取其前三維的 feature，並依據一組照片中那些 feature 的 value 切成 10 個 bins，最後我們對於同一個 bin 中的 feature point，做 exhaustive matching。

這個方法的好處，主要在於加速，實際上的 performance 和 exhaustive matching 差不多。



3. Cylindrical Warping

將座標系統從 x-y 座標轉換到圓柱座標，利用公式

$$\text{New_y} = f * y / \sqrt{x^2 + f^2}$$

$$\text{New_x} = f * \text{atan}(x / f)$$

可得到新的 x y 座標，並將原圖的 pixel value 填到新的座標上。

4. RANSAC:

設定一個 iteration number，對於每一次 iteration，我們假設某一組 pair 的 offset 是完全正確的，接著對於其他組 pair apply 這個 offset，並計算 inlier 的數量，並在每一次的 iteration，紀錄有最多 inlier 數量的 offset，當作我們最後的 offset。

5. alignment

在得到每兩組圖之間的 offset 後，我們要將圖依據這個 offset 值拼起來。因為 offset 還包含上下位移，所以困難點在於，要算出每個新加進來的圖要怎麼跟前一張圖排列在一起，也就是新的圖上下需要填補多少黑區塊才能跟前面的圖的大小一致。

6. Blending

我們使用兩種 blending 方法

A. Linear Blending

先求出兩張圖重疊的區域，給予兩張圖的 column 不同的 weight，從 1 -> 0，但是要注意的是，因為 warping 過後，圖片會有黑邊，所以如果第一張的圖片的 1~10 個 column 有黑邊，該 column 的 weight 就設為 0，再將第二張圖相對應的 column 的 weight 設為 1，如此即可避免黑邊的效應。



B. BONUS Multi-band blending

此方法主要是想要保留不同 scale 下的高頻資訊。

首先，先取出兩張影像重疊區域，再將這兩張影像分別做 gaussian pyramid (本次作業中使用的 variance = 13, layer = 10)，取完 pyramid 後，將前一層的 pyramid 放大兩倍與該層相減，可得到 Laplacian pyramid (laplacian pyramid 的最後一層即為 gaussian pyramid 的最後一層)。

接著建造新的 laplacian pyramid，建造方法為第一張圖的 laplacian pyramid 與第二張圖的 laplacian pyramid 做 linear blending。

再來一層層還原出 original image，還原方法為 $G_n = L_n + \text{expand}(L_{n-1} + G_{n-1})$ ；原文並沒有做 linear blending，而是直接取兩張圖的左右 laplacian pyramid 拼在一起，但是實作的結果發現在邊界沒有對好的地方這樣做不連續會很明顯，所以才改成 linear blending。



7. Bonus Recognizing panorama

給定一組順序被打亂的照片，我們的目標是把他們串接的順序找出來。
作法如下：

首先對於所有照片找出 feature point，接著對於所有 pair，做 feature matching，再跑 ransac，而在 recognizing panorama 中，跑 ransac 時，可以同時計算 inlier: n_i 以及在 overlap 區域中的 outlier: n_f ，在取得 n_i 以及 n_f 後，我們照著 paper 的做法，認定一組 pair 為 correct match if and only if $n_i > 5.9 + 0.22 * n_f$ ，而若有多張照片符合這個條件，我們挑選 n_i 最大的那個。

8. BONUS Seam Carving

在製作完全景圖後，因為有些黑邊需要被截掉，所以圖片會變扁，為了讓圖片變為原本高度，所以我們使用 **Seam Carving** 的方法來還原回原本高度。

此方法可以智慧地找出要如何縮放一張圖片而不會影響原本的圖片結構。方法為利用 **dynamic programming** 在圖上找出一條能量最低的 **path**(如圖中紅線部分)，新增一個 **row**，將此 **path** 以上的點都往上移一格，並且將這條 **path** 的顏色填為上下兩隔的平均，如此即可拉長圖片。此方法亦可透過尋找縱向的 **path** 來加寬圖片，但本次作業我們只需使用縱向加寬。



9. Result

