

HW2 Image Stitching

R09922188 曾泓硯, R09922089 蔡忠翰

May 6, 2021

1 Introduction

在這一次的作業中，我們的目標是用重疊的影像拼湊出一張全景圖。為了能夠正確地拼接影像，我們用 Harris 找出每一張圖中的特徵點，利用 SIFT 來敘述特徵點的資訊，譬如：角度、強度，接著配對兩張圖片的特徵點，透過 Homography Estimation 就能夠將影像兩兩一組拼接起來，最後將兩張影像的邊界做一些處理，使得影像自然地疊合。

2 Implemented Algorithms

- Cylindrical Mapping
- Harris Feature Detection
- SIFT Feature Description
- RANSAC
- Linear Blending
- Multi-band Blending (*bonus)

3 Block Diagram

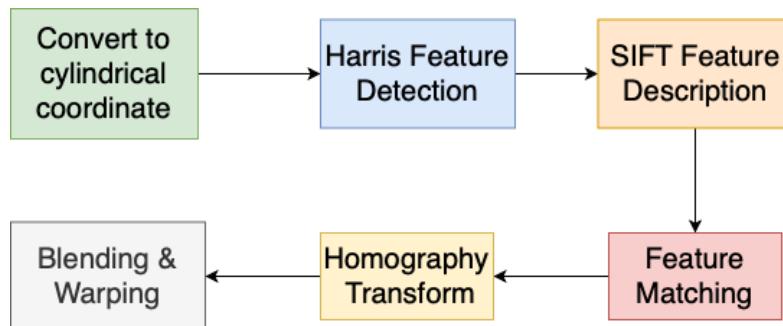
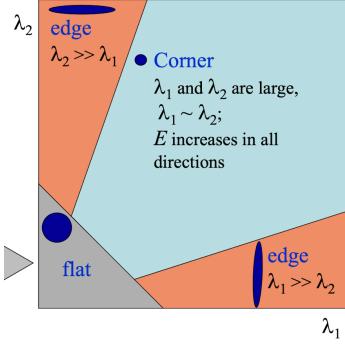


Figure 1: Our work flow

4 Implementation Details

4.1 Harris Feature Detection

我們使用了 Harris Detection 來當作我們找特徵點的方法。首先，我們會先利用濾波器來算出影像在縱向以及橫向的二次微分數值 ($\frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial y^2}, \frac{\partial^2 f}{\partial x \partial y}$)。我們預期特徵點在縱向以及橫向要有一定的變化量，而這件事情可以藉由特徵值的大小來判斷，計算的公式如 (Fig. 2(b)) 所示。我們可以發現當 $\lambda_1 \lambda_2$ 數值差不多且大於一定的門檻時，我們會將它定義為特徵點，如 (Fig. 2(a))。



(a) Harris Corner Detection

$$R = \det \mathbf{M} - k(\text{trace } \mathbf{M})^2$$

$$\det \mathbf{M} = \lambda_1 \lambda_2$$

$$\text{trace } \mathbf{M} = \lambda_1 + \lambda_2$$

(k - empirical constant, $k = 0.04\text{-}0.06$)

(b) Harris Corner Response

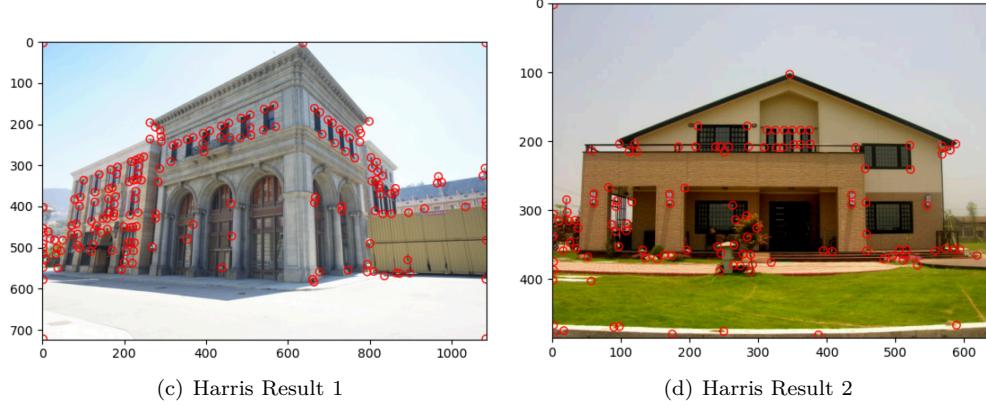


Figure 2: Harris Detection Result

4.2 SIFT Feature Description

我們利用 SIFT 的方法來敘述特徵點，這一部分可以分成 Orientation assignment 以及 keypoint descriptor。首先，我們透過影像的數值來計算出那個點的強度以及角度，接著我們會以一個區塊為單位，利用強度當作加權，算出此區塊內的角度應該是如何，以此來當作特徵的方向；此外，如果我們發現總和第二大的方向超過總和最大的方向的八成，我們會為此特徵點多分配一個特徵方向 (Fig. 3(a))。至於 keypoint descriptor 的部分，我們會根據特徵方向切出一個 16×16 的方格，接著以四格為一個單位，再算出方向一次，拿來當作他的 feature description，最後得到一個 128 維度的向量。

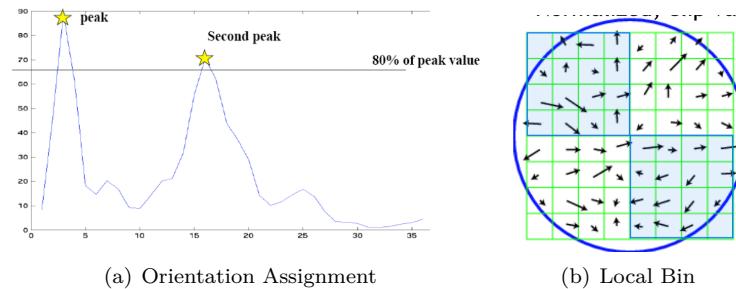


Figure 3: SIFT Orientation & Local Bin

4.3 Feature Matching

我們利用了 Harris 來偵測特徵點以及利用 SIFT 來描述特徵點，接著我們需要設法將特徵點對齊。我們將敘述特徵點的向量相減後做 l_2 norm，我們會希望這一對特徵點會跟其他特徵點相差很多，因此設定一個門檻來保證與其他特徵點對有一定的差距，藉此來確定這是真正的特徵點對 (Fig. 5)。

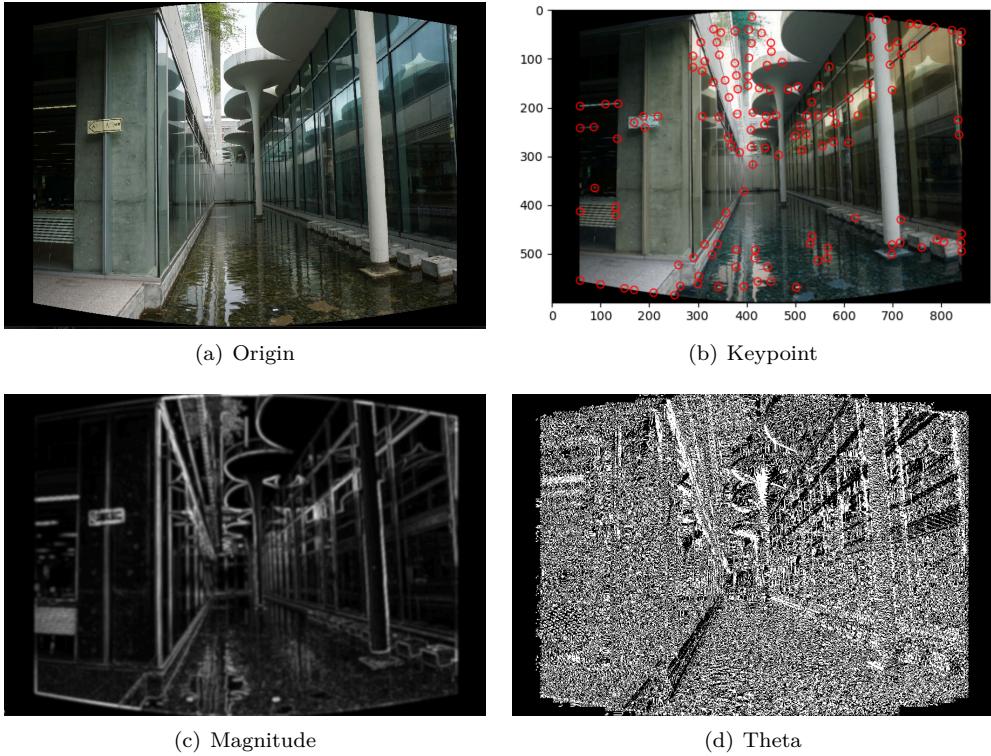


Figure 4: Orientation Result

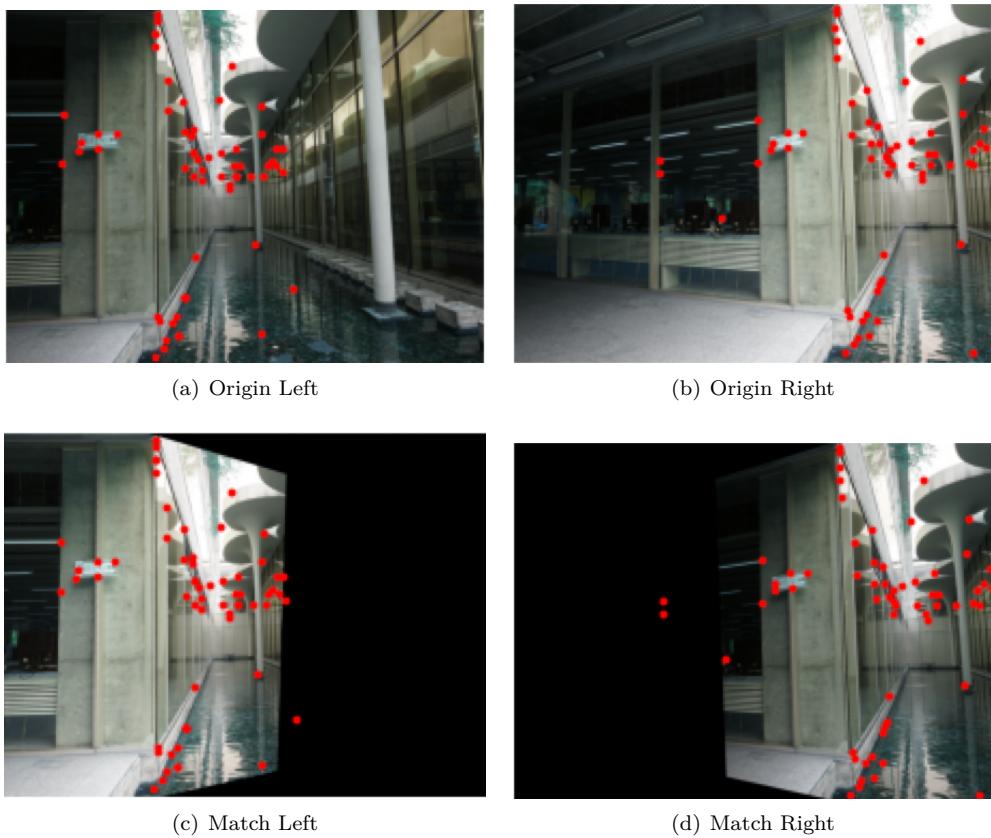


Figure 5: Feature Matching Result

4.4 Homography Estimation and RANSAC

在 Feature Matching 過後，我們有了每一組特徵點的座標對應關係，我們便可以利用這些資訊來推算兩個影像之間的對應關係，在這之中牽涉到我們怎樣為圖片之間的座標轉換建立數學模型。我們總共有實作兩種：一種是投影矩陣，有八個自由度；另一種則是平移矩陣，僅有兩個自由度。乍看之下，選擇自由度很高的模型是比較好的，因為能夠擬合出更多關係，但與此同時也很容易受到前面步驟估算誤差的影響。在我們自己的實驗中，因為圖片幾乎都是利用角架拍攝，且沒有劇烈的景深變化。因此利用平移矩陣就已經足夠估算出好的結果。

不過，即使將數學模型定義到只有兩個自由度，如果我們直接利用所有配對好的特徵點去計算，可能還是會受到前面步驟的錯誤所影響。因此我們可以利用隨機的方式，去挑選出足夠解出一個模型所需要的特徵點組合，接著利用估算出的矩陣對其他點進行轉換以後，看估算值與實際值的差距是多少。如果一個矩陣能夠使在閾值內的特徵點組合越多，就表示這是個更準確的估計值，如此一來就可以避免掉一些計算或配對錯誤的特徵點。

4.5 Multi-band Blending

在將影像推算過座標對應關係並轉換以後，因為相機拍攝的參數不同或是場景的局部差異，在兩張圖的交界處會有兩種資訊，如果只取用任意單一張的圖片資訊，都會導致視覺上突兀的跳變。原先我們實作的是簡單的 Linear Blending，即是如同講義上所述，在交界處線性地混合兩者的像素值。而這種方法在邊界如果有較多高頻細節，且 Homography 無法使圖片完美貼合時，會有些許鬼影的感覺 (Fig. 6(a))。

而原論文是採用 Multi-band Blending 來進行邊界混合，其流程簡單來說，就是先針對兩張影像的交疊處，各自建一個 Gaussian Pyramid 以後，再利用該金字塔推算出 Laplacian Pyramid (Gaussian Pyramid 上下兩層的差)，接著將兩張影像的 Laplacian Pyramid 混合成一個金字塔以後，在 Gaussian Pyramid 中取最小張的圖片混合並放大以後，加回混合後的 Laplacian Pyramid 中最小張的圖片；再來，將上一階段的結果放大以後與混合後的 Laplacian Pyramid 中第二小張的圖片疊加，以此類推，直至回到原先的大小為止。如此一來，就可以做到類似使低頻混合，但保留高頻的效果，結果如 (Fig. 6(b)) 所示。



Figure 6: Blending Result

5 Result



Figure 7: Panoramas Results



Figure 8: Cropped Panoramas Results

6 Conclusion

這一次的作業相當有挑戰性，尤其是當我們以為自己做出足以敘述特徵點的程式時，卻在後面的 Homography Estimation 時跑出奇怪的結果，除了改變 Homography 的自由度外，我們還回頭去調整了很多次參數，最後才發現原來是 feature description 的地方出了一點錯誤，導致特徵點彼此無法順利對齊。此外，我們也有拍攝了一組在半室內的照片，當我們將影像投影到圓柱上後，發現室內室外的交界處難以接合，得到的點數過少，也導致全景圖遲遲無法完成，也因此最後我們拍攝的照片選擇的是戶外的場景。

References

- [1] <https://lerner98.medium.com/implementing-sift-in-python-36c619df7945>
- [2] <https://kushalvyas.github.io/stitching.html>
- [3] <https://github.com/cynricfu/multi-band-blending>