

Digital Image Processing
And Computer Vision
Final Project Report

Image Alignment and Application

工科所 N96104161 劉宜珊
數據所 RE6101037 陳劭寧

中華民國 一百一十一年 一月 十一 日

目錄

一、緒論.....	2
(一)摘要	2
(二)動機.....	2
(三)目的	3
二、執行步驟	4
(一)Image alignment.....	4
(二) Application 1	6
(三) Application 2	6
三、結果展示與分析	7
(一)基本輸出	7
(二) Application (1)	8
(三) Application (2)	9
四、結論與未來展望	10
(一) 結論	10
(二) 未來展望	11

一、 緒論

(一) 摘要

Image Alignment 在許多大型的影像處理系統都經常使用，例如：即時影像追蹤、衛星影像觀測、醫學影像臨床觀察……等。Image Alignment 是指找出兩張影像之間的變換關係，如平移、旋轉、縮放，經過變換之後使兩張影像中相同的部分可以重疊，它可以讓一張可能在拍攝過程中歪掉的照片轉正，將圖像扭曲旋轉使他和另一個圖可以很完美的對齊，且與另一個圖一樣的方向。影像之間的變換關係可以用一個矩陣來表示，所求得矩陣的未知數數量越多，代表兩影像之間的關係越複雜、越難對準。為簡化問題，在拍攝影像時會盡量避免同時平移、旋轉或縮放，以簡化對準的難度。

Image Alignment 能夠解決許多問題，看到目前許多應用便可知，因此我們決定也以此技術來解決生活上遇到的困擾，甚至將這個技術運用在未來若無人商店普及時能夠有將商品辨識準確率提升的貢獻。

(二) 動機

在拍攝物體時，有時可能會因趕時間或是沒辦法在能夠拍出清楚影像的位置拍攝，這時就會造成想拍攝的目標物有歪斜的問題。以自身碰到的情況來說，可能因當下沒有掃描的工具，對方又急著要看文件但電子檔丟失，所以只好用拍攝的方式給對方看文件，但他人接收此翻拍卻又因拍攝得歪斜而

使文件看不清楚。以自身生活上遇到的問題為出發點，我們決定以 Image Alignment 作為這次的期末 project 主題。

(三) 目的

我們期望能夠將 Image Alignment 應用於日常生活中，在此我們做了兩個生活上的應用，這兩個應用主要從自身所碰到的狀況與商家可能碰到的困難去著手。第一個應用為在貨架上的辨識，此應用可以幫助商店用以圖搜圖的方式，尋找該商品的位置，以節省時間以及人力成本的優勢下，進而吸引更多客戶購買，也可藉此提升銷售率，甚至可以在日後無人商店普及化時對商品辨識準確率的提升做出一點貢獻。第二個應用為處理翻拍之後照片中目標物歪斜的問題，此應用能夠順利幫助我們解決收到別人翻拍歪斜的文件的困擾。

二、 執行步驟

(一) Image alignment

1. Convert images to grayscale

用 cv2 裡的 cvtColor 函數將彩色圖片轉換為灰階圖片。

2. Detect ORB features and compute descriptors

使用 detectAndCompute 可以偵測出特徵點並計算 descriptors，通常計算

Homography matrix 至少 4 個配對點就夠了，但一般來說會配對出成千

上萬個特徵點，所以這邊使用 MAX_FEATURES 來限制數量。而在

test_from_1_to_7.py 中，MAX_FEATURES 設定為 2000 ；

test_from_8_to_11.py 中，MAX_FEATURES 設定為 5000 ；

test_from_12_to_13.py 中，MAX_FEATURES 設定為 10000

3. Match features

這裡使用 cv2 裡的 DescriptorMatcher 函數來找到兩圖中匹配的特徵

點，而匹配的方法使用漢明距離來衡量兩個 descriptors 的相似度。

4. Sort matches by score

將匹配度(matches score)由大到小排列以便日後提取較好的 matches。

5. Remove not so good matches

將 good matches 設定為 matches 總數乘上 GOOD_MATCH_PERCENT，

過濾掉乘積(numGoodMatches)之後的 matches，以留較匹配的一小部分。

- 在 test_from_1_to_7.py 基本輸出的 code 中，

GOOD_MATCH_PERCENT 設定為 0.06。

- test_from_8_to_11.py 翻拍紙張校正後黑白二分化的 code 中，

GOOD_MATCH_PERCENT 設定為 0.03。

- test_from_12_to_13.py 應用在貨架上辨識的 code 中，

GOOD_MATCH_PERCENT 設定為 0.012。

6. Draw top matches

使用 cv2 裡的函數 drawMatches 去畫出輸入圖(input_img)和參考對比圖(ref_img)所匹配的特徵點圖(匹配的特徵點用線連起來)，並存入資料夾中。

7. Extract location of good matches

將過濾後得到的 good matches 提取出他們的 index 位置。但匹配完後還是有很多錯誤匹配的特徵點，所以下一步要用一個強韌的算法來計算 Homography matrix。

8. Find Homography

上一步產生的匹配的特徵點不是 100%正確的，就算只有 20~30%的匹配也是常見。所以這裡使用 cv2 裡的 findHomography 函數，它使用一種被稱為隨機抽樣一致算法(Random Sample Consensus)的技術，可以在大量匹配錯誤的情況下計算 Homography matrix。

9. Use Homography

有了 Homography matrix 後，就可以把一張圖片的所有像素映射到另一個圖片。而這裡用 cv2 裡的 warpPerspective 函數用來完成這個功能。

(二) Application 1

將貨架上歪斜的飲料罐，針對特定的兩項飲料去做校正

1. Input reference image

要找的目標物圖片

2. Matched

配對出兩者的 feature point

3. Get result via location

找出配對點較集中的地方並推測出目標物的位置

(三) Application 2

主要可以處理所翻拍的照片，

1. Input

輸入翻拍的照片

2. Align Images

執行一整個 Image Alignment 過程

3. Something need to fix

把圖片轉成黑白之後，再以 pixel 值=128 為分界點，分成黑色和白色

三、 結果展示與分析

(一)基本輸出

[結果]

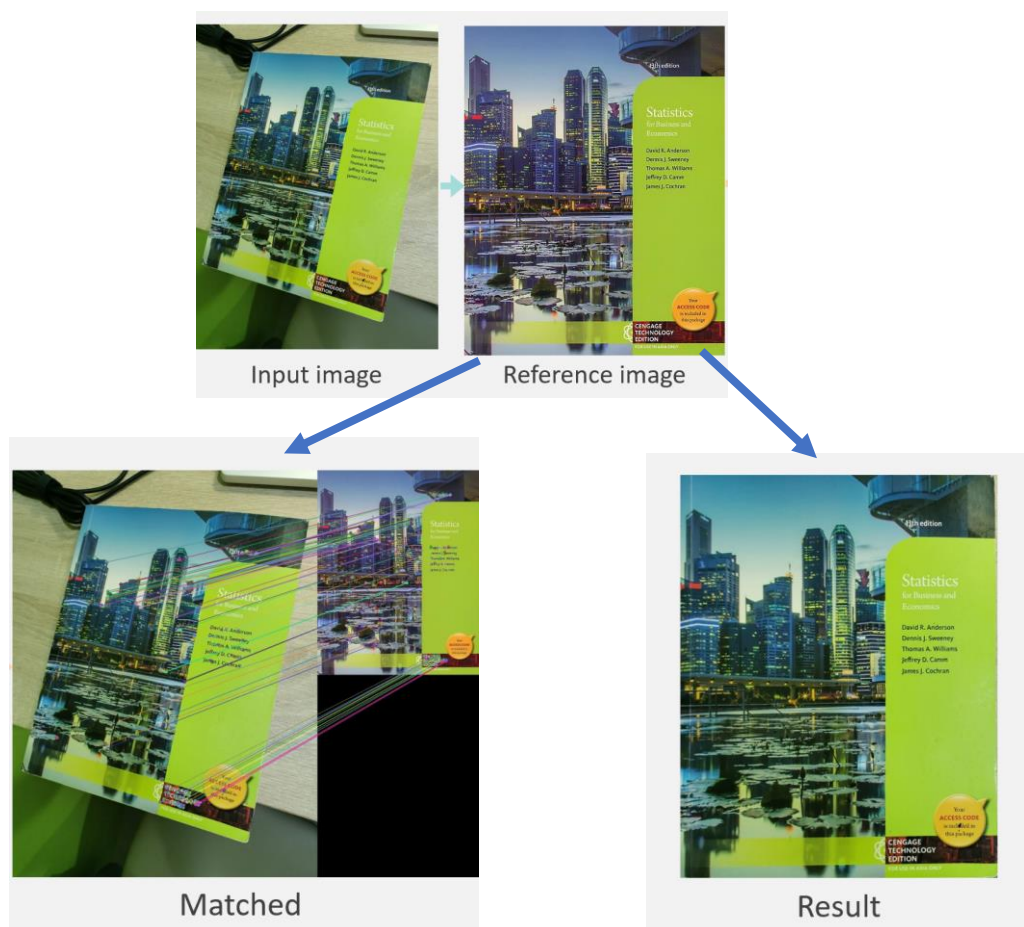
檔案：test_from_1_to_7.py

主要為基本輸出的 code，原圖(Input_img1~ Input_img7)

alignment 後的特徵點匹配連線圖(match1~ match7)

放在 match 資料夾裡，匹配後的結果圖(result1~ result7)

放在 result 資料夾裡。圖片總共有 7 張，這裡以 input_img4 為範例呈現結果圖。



[分析]

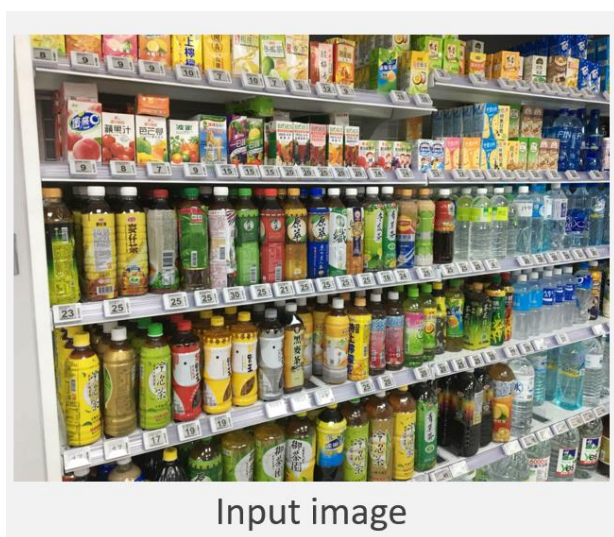
原圖歪斜的相當明顯，而所拍攝到的背景桌子也使得個目標物變得比較小且不清楚，經過 image alignment 後，匹配的結果圖與參考圖 (ref_img4) 用肉眼來看幾乎無任何差別，結果相當不錯。

(二) Application (1)

[結果]

檔案：test_from_12_to_13.py

主要應用在貨架上辨識，原圖(Input_img12、Input_img13) alignment 後的特徵點匹配連線圖(match12、match13)放在 match 資料夾裡，匹配後的結果圖(result12、result13)放在 result 資料夾裡。



[分析]

與原本的基礎輸出結果相比，匹配後的結果圖很明顯的與對比圖有些許的差別，雖然有將原本歪斜的飲料瓶調正一點，但還是沒有辦法做到像基本輸出的範例圖一樣，肉眼難以看出與參考對比圖 `ref_img` 的差異。推測可能是因為原始圖片畫質不夠好，進而導致貨架上物品的畫質更小，進而配對出較差結果。

(三) Application (2)

[結果]

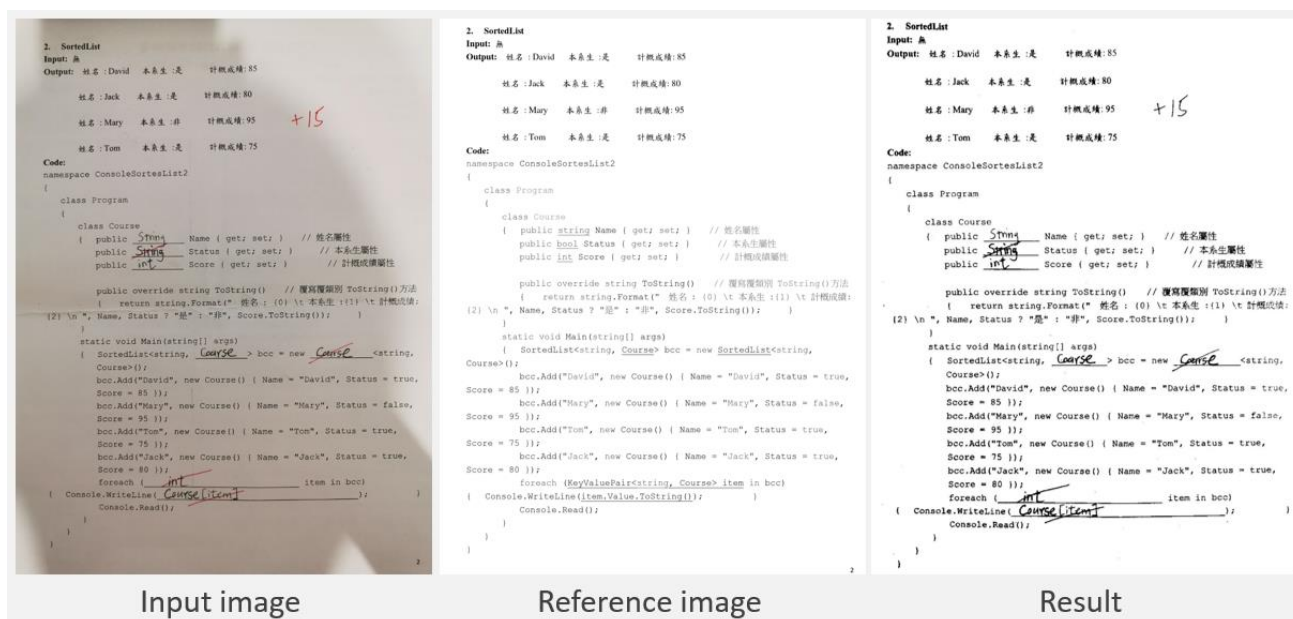
檔案：`test_from_8_to_11.py`

主要為翻拍紙張校正後黑白二分化，原圖(`Input_img8~Input_img11`)

`alignment` 後的特徵點匹配連線圖(`match8~match11`)放在 `match` 資料夾

裡，匹配後的結果圖(`result8~result11`)放在 `result` 資料夾裡。圖片總共有

4 張，這裡以 `result 9` 為範例呈現結果圖。



[分析]

Image Alignment 後的結果 result9 與 reference image 十分相近，除了考卷上寫答案的字跡和紅筆所批改的痕跡不同以外，而原本 input image 因拍攝而產生的陰影和考卷的摺痕在經過轉換後，肉眼也幾乎看不見了，甚至文字的顏色還更深更清楚了。

四、 結論與未來展望

(一)結論

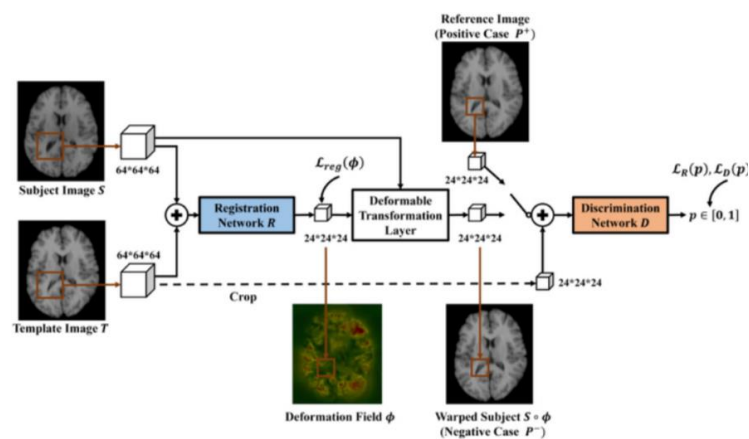
Image Alignment 成功解決了我們在生活中所遇到的困難，輸出的結果都算不錯，要注意的大概就是照片畫質，含有多個物體的照片和只有單個物品照片，單個物品的效果又會比較好，因此在貨架上辨識應用上

效果沒有翻拍應用來得好。所以未來會著重在貨架上辨識應用的優化，可能的方法為針對貨架上的圖進行第二次 Image Alignment，或是再找出其他深度學習的方法去做調整和優化。

(二)未來展望

參考 Adversarial Similarity Network for Evaluating Image Alignment in Deep Learning based Registration 這篇論文去做 Image Alignment 的優化，此篇論文方法為根據 discrimination network 的反饋去進行配準網路 (registration network) 的訓練，而 discrimination network 旨在判斷一對 Image Alignment 配對圖像是否足夠相似，有點類似 GAN 生成器和判別器的概念。

整體架構主要分為兩個部分：(1)配準網絡 R (registration network) (2) 判別器網絡 D (discrimination network)，結構圖如下圖所示：



在 R 網絡之後是一個 Deformation Transformation Layer 用於對 Moving image 進行變換，得到 Warped image。網絡 D 用於判斷輸入的圖像對是

否相似。正樣本表示很好的配準，負樣本表示配準不好。

以上為參考論文的概念流程，期望之後細細研究此篇論文將 Image Alignment 進行優化後，能夠將貨架上辨識應用的效果變得更好，並且能夠解決商家的困擾和更多生活可能遇到的各種問題。