

數位影像處理

Digital Image Processing

期末專題 Final Project

Group 10

R11945054 謝展睿

R11945053 鄭詩妮

R11921038 江讀晉

2022/12/18

Outline

- Introduction
- Methodology
- Experiment and Results
- Conclusion

Introduction

這學期的期末專題，我們第 10 組所選擇的主題為「Seamless Focal-Stack Refocusing」，其動機為手機所拍攝的變焦影片或相片集（focal-stack）會在變焦過程中，出現畫面放大或縮小的情況。因此，本次專題的目標為將上述提及的變焦相片集，透過 image rectification 之後，達到畫面大小一致的效果，並且可以在螢幕（本組僅使用電腦，未於手機展示）上任意切換對焦之位置。

為了實踐上述目標，我們使用 iPhone 8 以及 iPhone 13 收集所需要的 raw video，再將其轉換成一張張的 frame。至於中段處理 image rectification 的部分，為了找出較為理想的結果，我們嘗試了三種方法：ORB feature & brute-force matching、Optical flow & brute-force matching、resizing and cropping，並從中找出表現最好的方法。之後再進行…。整體的流程圖如下方 Figure 1 所示。

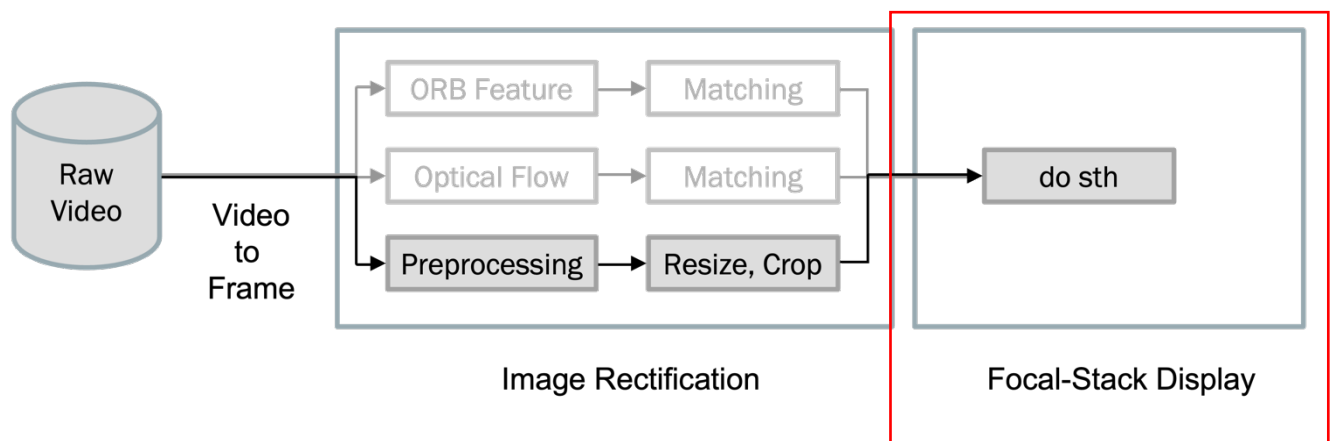


Figure 1 本次專題的流程圖

Methodology

1. Image Rectification

以下依序介紹我們所嘗試的三個方法：ORB feature & brute-force matching、Optical flow & brute-force matching 和 resizing and cropping。

| Method 1: ORB feature & brute-force matching

在一組 image pair 中，要找出 2D-to-2D 的 feature correspondence，大致可分為以下的三個步驟[1]。

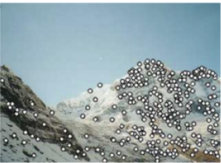
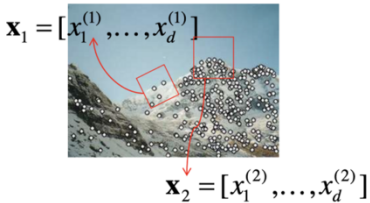
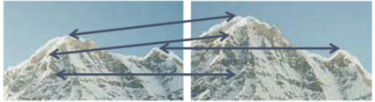
Detection	Description	Matching
識別出有興趣的 points	提取出有興趣的 points 周遭之 vector feature descriptor	決定兩張照片中 descriptor 的 correspondence
		

Table 1 feature correspondence 簡介

Detection 和 description 的步驟中，我們所使用的是 ORB (Oriented FAST and Rotated BRIEF) [2]，它是 FAST keypoint detector 和 BRIEF descriptor 的整合演算法。由於 ORB 效率較課堂上提及的 SIFT 高，因此選擇使用 ORB。至於 matching 方式則使用 brute-force 方法，找出 evaluation 最高的 point pairs。

Figure 2 呈現 ORB 加上 brute-force matching 的大致流程。此外，將兩張影像 matching 的點分別以 pts1 和 pts2 回傳，以方便後續步驟計算 homography。

ORB_BFmatching: input - image1, image2 / output - points1, points2
1. Initiate ORB detector and BF matching object (via OpenCV)
2. Find the keypoints and descriptors with ORB
3. Match descriptors
4. Sort the matching results in the order of their distance.
5. Use query index and train index separately to find matching points for image1 and image2 → points1, points2
6. Return points1 and points2

Figure 2 feature matching 流程概述

找出 local feature correspondence 之後，使用這些 matching points 找出 homography，將當前處理的 frame warp 至對齊前一張 frame，以達到畫面大小一致的效果。Homography 的示意圖

如 Figure 3 所示，公式如下：

$$\lambda_i \begin{bmatrix} u_i' \\ v_i' \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}$$

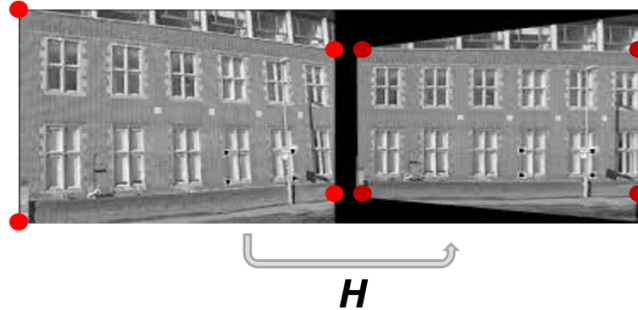


Figure 3 透過 homography，將左圖個座標轉換至如右圖所示

| Method 2: Optical flow & brute-force matching

除了 Method 1 提及的 ORB feature 之外，我們也嘗試使用 Optical flow[3]的方式。Optical flow 為兩連續 frame 之間，移動物體所產生的明顯運動軌跡。透過此種方式，我們也可以找出 feature correspondence，並做進一步計算。

Method 2 大致的步驟和上一小節提到的相同，不同之處為將 track points 作為兩 frame 之間的 correspondence，並計算 homography 來作 warping。

| Method 3: Resizing and Cropping

根據課堂給的 reference 投影片中，image rectification 使用 resizing & cropping 的方法，找出可以算出兩 frame 之間相差的 L2-norm 之最小 resize ratio，公式如下：

$$r^* = \underset{r}{\operatorname{argmin}} \| \operatorname{ResizeCrop}(\operatorname{edge}_{k+1}, r) - \operatorname{edge}_k \|_2$$

其中 r 為 resize ratio， $\operatorname{edge}_{k+1}$ 、 edge_k 為經過前處理所產生的 frame pair。

Resizing & cropping 的方法較容易實踐，且可以保持視角的一致性。不同於 Method 1、Method 2 使用 feature correspondence 的方式，此處使用 grid search 的方式找出最適合的 ratio size。

此外，由於 iPhone 8 的感光應變能力較弱，所收集的 frames 在 refocus 時可能造成畫面繪圖染過曝，如果直接使用原本 image 的 rgb 值去找做適合的 ratio，可能會造成嚴重誤差。因此，我們先將每個 frame 進行 preprocessing，使用 Canny algorithm 求出 edge，再進行運算。如此一來，能夠得到較為穩定的 resize ratio 變化。

| Summary

以下呈現 image rectification 的 pseudo algorithm，主要的計算皆使用 OpenCV 函式：

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	Image_Rectification: <img_last <math="">\leftarrow img0 for_loop from path1 to pathN, do <img_curr <math="">\leftarrow imread(path) # Method 1 pts_last, pts_curr \leftarrow ORB_BFmatching(img_last, img_curr) H \leftarrow findHomography(pts_last, pts_curr) <img_curr_new <math="">\leftarrow warpPerspective(img_curr, H) # or Method 2 pts_last, pts_curr \leftarrow optical_flow(img_last, img_curr) H \leftarrow findHomography(pts_last, pts_curr) <img_curr_new <math="">\leftarrow warpPerspective(img_curr, H) # or Method 3 <img_curr, <math="" ratio="">\leftarrow resize_crop(img_last, img_curr) <img_last <math="">\leftarrow img_curr_new end </img_last></img_curr,></img_curr_new></img_curr_new></img_curr></img_last>	# Capture new frame # ORB descriptor and matching # Compute homography # Image warping # Optical flow find track points # Compute homography # Image warping # Use resizing and cropping to find the ratio of interest # Update last image
---	--	---

Table 2 image rectification pseudo code and comment

綜合這三種方法的表現，目前僅以目測方式去判斷，為 resizing & cropping 的表現看起來最為穩定，如 Figure 4 所示 [4]。因此，之後在 focal-stack 的展示，我們皆使用 resizing & cropping 去達到畫面大小一致的效果。

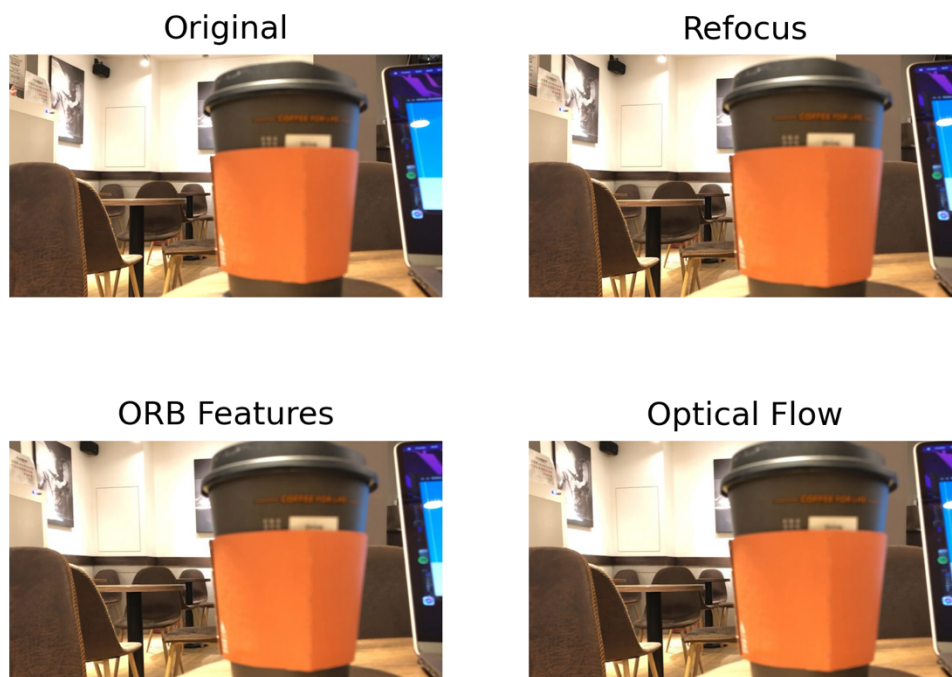


Figure 4 Image rectification 不同方法的結果比較，ORB features 和 optical flow 會放大且變形

Experiment and Results

1. Brief Demo: Stacks of Frame, Edge and Resize Ratio

此次專題中，我們一共測試了三組 frame set。Figure 5 和 Figure 6 呈現出對焦於前景以及背景的 edge 和 ratio 變化 [5, 6, 7]。當對焦於前景時，前景物體的 edge 輪廓較為明顯；對焦於背景時，後方物體的 edge 明顯，前方物體則是因為失焦幾乎偵測不出原本的 edge，且由於畫面縮小，內容物叫對焦於前景時多，因此可以些微放大的 resize ratio。

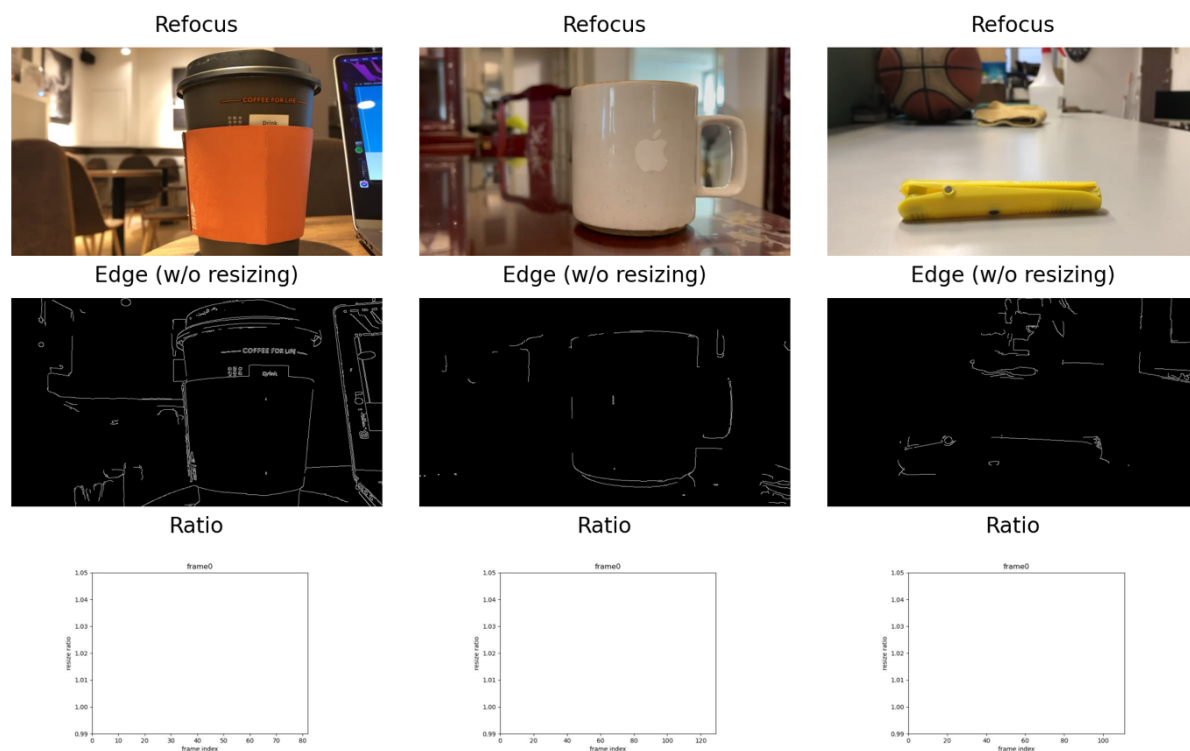


Figure 5 三筆 focal-stack 資料對焦於前景時的 edge，由於前景畫面較大，此時 ratio 尚未有變化

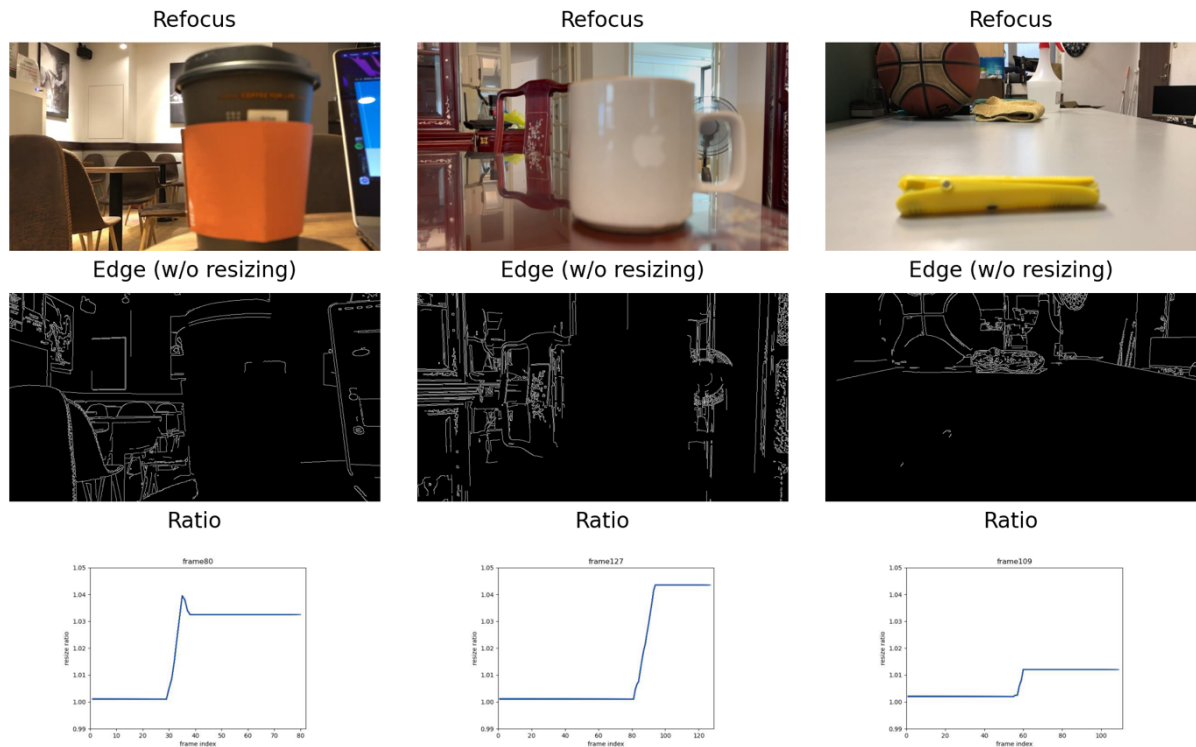


Figure 6 三筆 focal-stack 資料對焦於前景時的 edge 和 ratio 變化

Reference

- [1] Kristen Grauman. Local features: detection and description - UT Computer Science. 2011
https://www.cs.utexas.edu/~grauman/courses/trento2011/slides/Monday_LocalFeatures.pdf
- [2] Rublee, Ethan; Rabaud, Vincent; Konolige, Kurt; Bradski, Gary (2011). "ORB: an efficient alternative to SIFT or SURF". IEEE International Conference on Computer Vision (ICCV)
- [3] B. D. Lucas and T. Kanade (1981). "An iterative image registration technique with an application to stereo vision". Proceedings of Imaging Understanding Workshop, pages 121—130

YouTube Link

- [4] Comparison of Resizing and Cropping, ORB Feature and Optical Flow Method
<https://youtu.be/2grvsjDTN6s>
- [5] Brief Demo -- Apple Cup
https://youtube.com/shorts/_68isQSwong
- [6] Brief Demo -- Chalk
<https://youtube.com/shorts/ieKlgwOVs60>
- [7] Brief Demo -- Louisa
<https://youtube.com/shorts/nlWg8UIh3MQ>