

3DCV hw1

R12945040 生醫電資所 郭思言

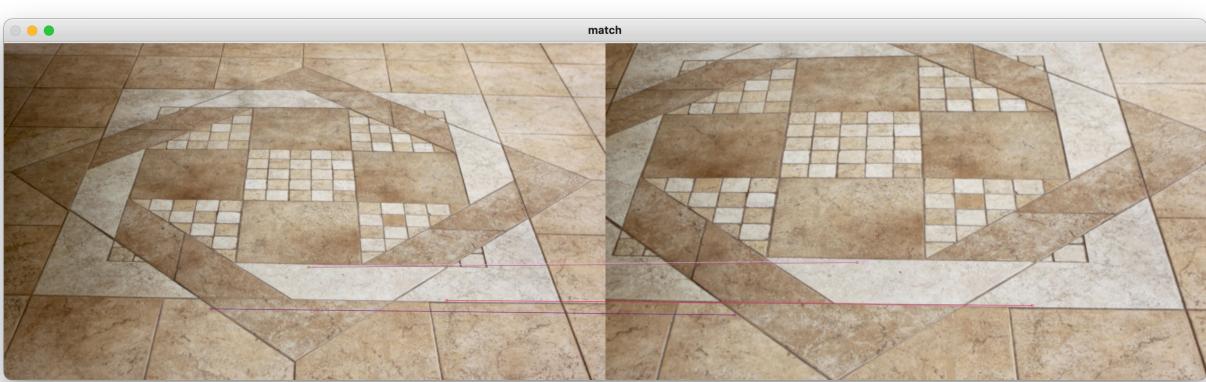
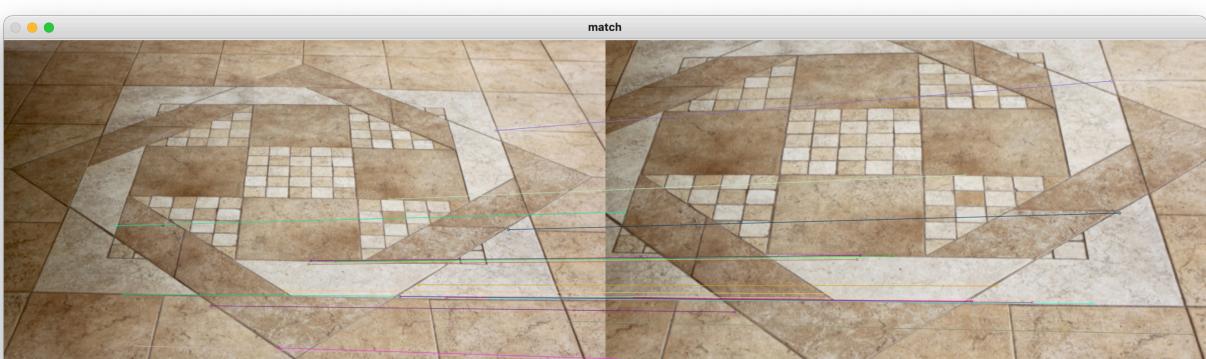
YouTube link: <https://youtu.be/Ewk-BmshZ5Y>

Problem 1: Homography Estimation

Q1-1 Feature Matching:

Image pair #1

Sample k correspondences:

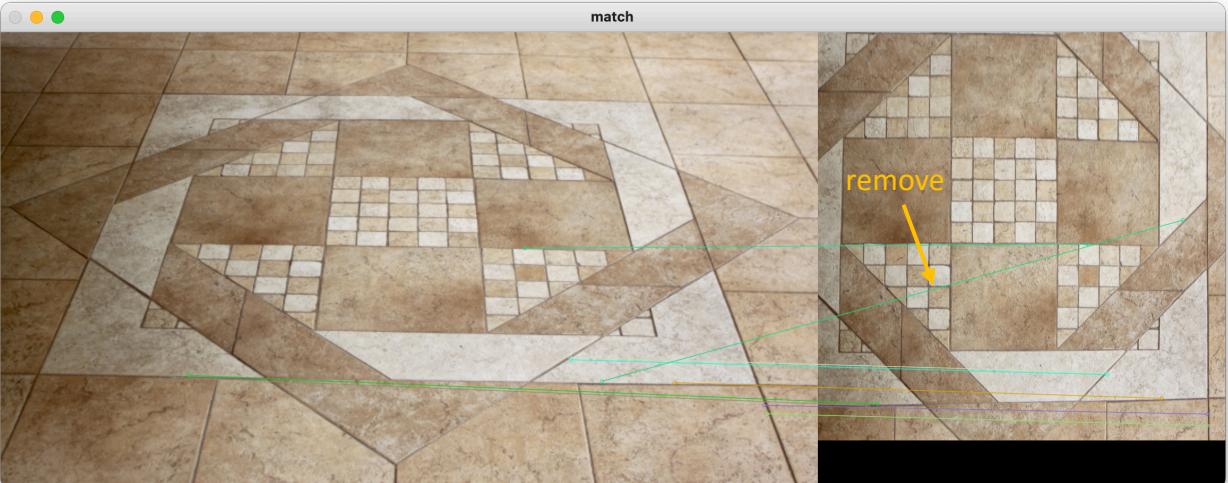
4	
8	
20	

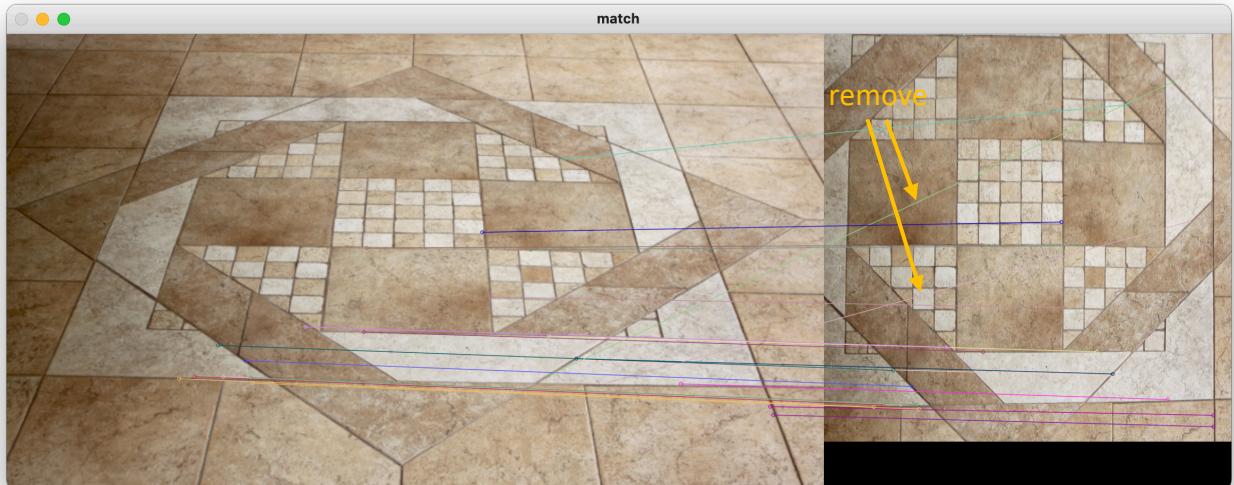
Compare the errors:

	DLT	Normalized-DLT
4	21.69864120537615	21.698641198654926
8	0.21408877462149595	0.2038160858668027
20	0.03385417587897089	0.03280810496979946

Image pair #2

Sample k correspondences:

4	
8	



Compare the errors:

	DLT	Normalized-DLT
4	43.85605459829871	43.856054599412076
8	1.8037924537398007	4.648393824073985
20	0.9230475002549413	0.5683018533707263

Discussion

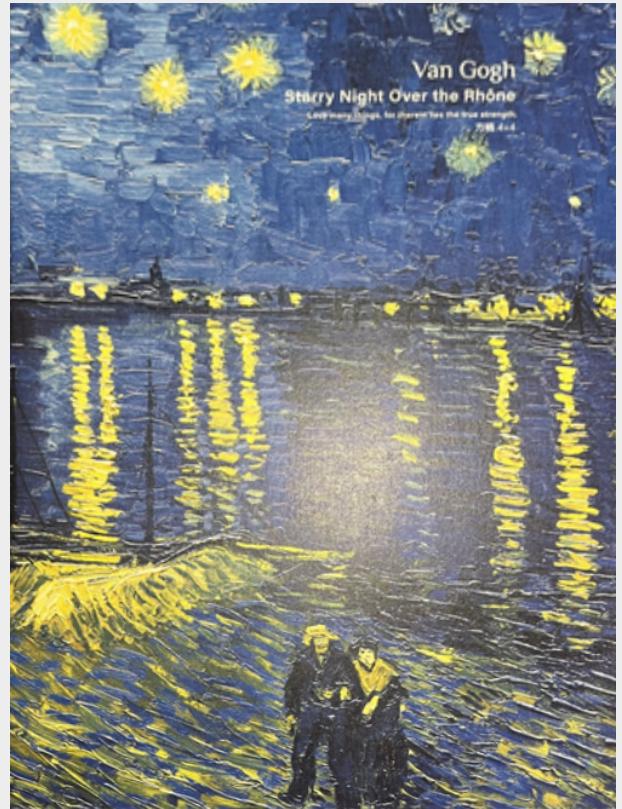
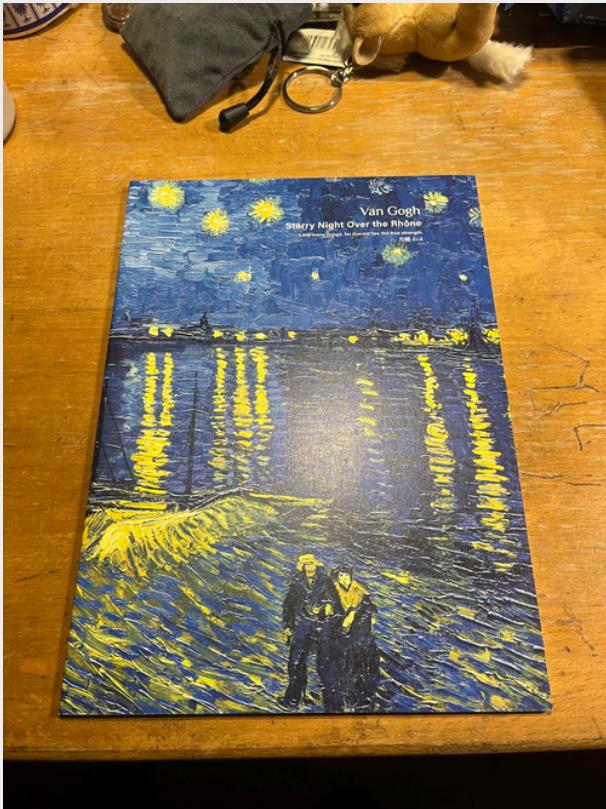
1. Image pair #1 is relatively simpler compared to Image pair #2. It can produce normal results without manual selecting the SIFT correspondences.
2. When using SIFT-based correspondence selection directly on Image pair #2, the results are not satisfactory. The error actually increases over time. Upon inspection, it was found that the 6th and 15th pairs were not correct. After manually removing them, the results became more normal.
3. Looking at two image pairs, selecting more correspondences to run the DLT algorithm can calculate a more accurate Homography, significantly reducing the error.
4. Looking at two image pairs, using normalization can slightly improve performance and slightly reduce the error. However, in Image pair #2, the error increased for 8 correspondences, which leads me to suspect that there are still some inaccuracies in the selected correspondences.

Problem 2: Document Rectification

Input document image

Rectified result

2-1



2-2



Briefly explain my method:

1. Each time this program is executed, the user needs to manually click to select the four corners of the target area in the input image (you can select the four points in any order).
2. After the selection is completed, the program will automatically calculate the Homography using the Normalized-DLT algorithm and warp the target area to a new image of the same size as the original image. This new image is then saved in the original directory.
3. The complexity of backward warping with bilinear interpolation
 - (1) Bilinear interpolation involves taking a weighted average of the four nearest neighbor pixels in the input image. For a single pixel this operation is typically O(1) (fixed number of operations: four pixel value reads and some multiplications and additions).

- (2) Therefore, the overall complexity of backward warping using bilinear interpolation is usually linear with respect to the number of pixels in the output image $O(N)$. (N: number of pixels)