UNIVERSITY of LIMERICK

OLLSCOIL LUIMNIGH

Group Name:	Kevin Burke (14155893)
	Paul Lynch (16123778)
	Ciaran Carroll (13113259)
	Qicong Zhang (16069978)
Module:	RE4017 Machine Vision
Lecturer:	Dr. Colin Flanagan
Project Title:	Harris Corner Detector
Year:	2017/18
Date:	24th April 2018

Introduction

The Harris Corner detector was developed and first introduced by Chris Harris and Mike Stephens in 1988 at the Alvey Vision Conference, in their paper *"A Combined Corner and Edge Detector"*. It is a technique, frequently used in computer vision algorithms, which detects corners from a given input image. A corner is defined as 'a point whose local neighbourhood stands in two dominant and different edge directions'.

There are many uses of this algorithm in the real world, for example in object recognition, video tracking and image stitching and alignment, which this project will focus on.

Aims

It is possible to reconstruct a single, unified image from two matching image 'pairs'. The aim of this project is to take sets of test images, shown in Figure 1(a) and (b), and stitch them together to reconstruct a partial original image.

This project will use the Harris Corner detection algorithm in order to identify the salient corners in each image. We will then attempt to identify the matching points and blend the two images into one another.

Left: Figure 1(a) Image pair (Arches)

Right: Figure 2(b) Image pair (Hot air balloon)

Implementation

In order to implement the image stitching application described in the aims, a number of steps must be followed. The Harris Corner detection algorithm should first be applied to both images. Similar points should then be matched and a translation can be found by means of Random Sampling Consensus (RANSAC).

Harris Corner Detection

The algorithm developed by Harris and Stephens follows these steps:

1. Edges in an image are found by convolving the derivative of Gaussian x & y kernels with sigma = 1, giving $I_x(x, y)$and $I_y(x, y)$
2. $I_x(x, y)^2$, $I_x(x, y)I_y(x, y)$ and $I_y(x, y)^2$ are derived for each point in the image.
3. These are then convolved with a patch filter, which is a Gaussian with sigma = 2, resulting in:

$$A = G_{\sigma 2} * {I_x}^2, \quad B = G_{\sigma 2} * I_x I_y\ , \quad C = G_{\sigma 2} * {I_y}^2$$

4. Matrix M is given by:

$$\begin{bmatrix} A & B \\ B & C \end{bmatrix}$$

   From this, R can be found by:

$$R = \frac{\det(M)}{tr(M)}$$

5. R can then be thresholded by performing nonmax suppression and returning the remaining points as Harris corners.

The results of this implementation on Python can be seen in Figure 2(a) and (b) for the first set of test images and Figure 3(a) and (b) for the second set of test images.

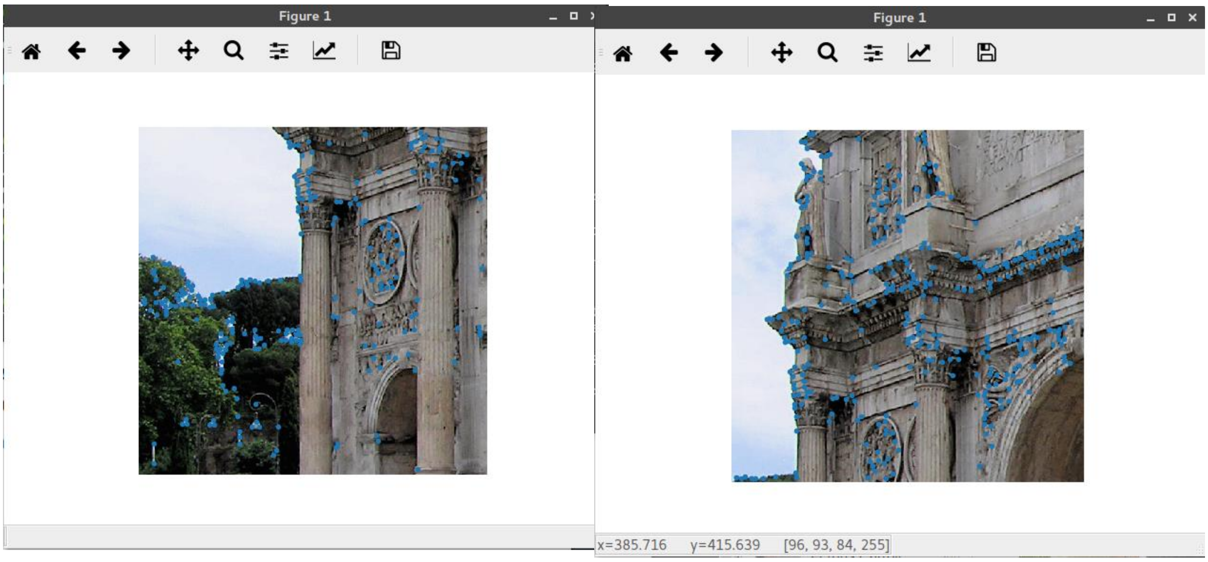


Figure 3(a) and (b) Harris points detected in Image 1

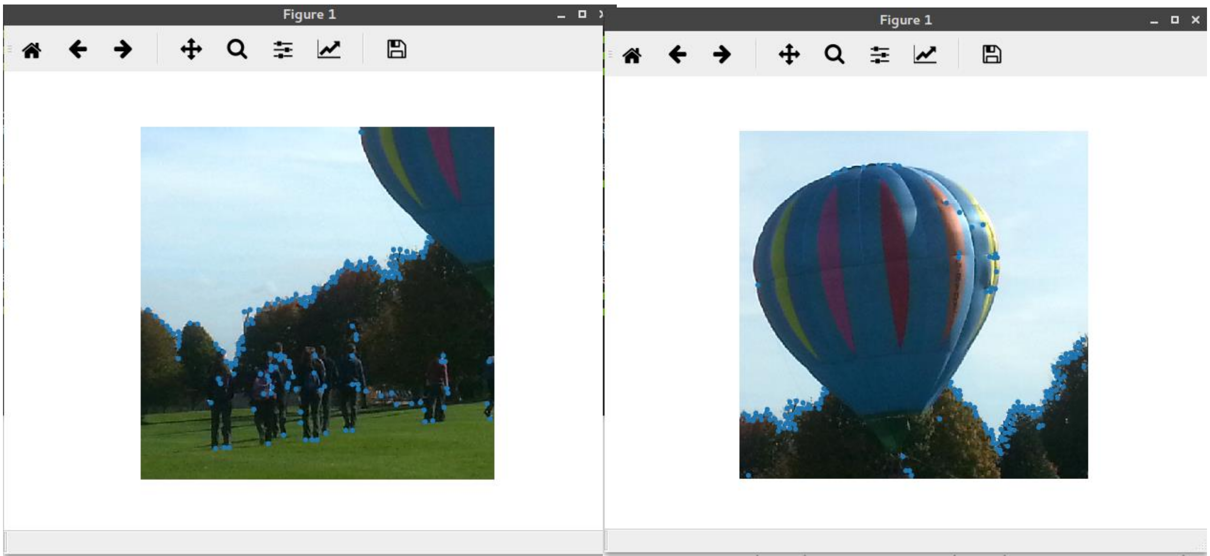


Figure 4(a) and (b) Harris points found in Image 2

The results for both pairs of test images are quite good. The majority of all points identified by the algorithm are in fact corners. It is already clear to the human eye the translation that should take place in order for both pairs of images to merge and stitch together.

There are many matching points between the pairs, but also some unique corners which do not appear in both images. The next step in the process is identifying which corners are 'matching' and how strong the match is.

Corner Matching

In order to match the similar corners in the image pair, the normalised cross correlation technique is used. This involves comparing the descriptor (neighbourhood around the point) of each Harris corner in the first image with every descriptor in the second image. If the match is greater than a 0.95 threshold, the two points are taken to be matching.

Figure 5(a) and (b) shows the results of matching similar corners on each of the test image pairs.

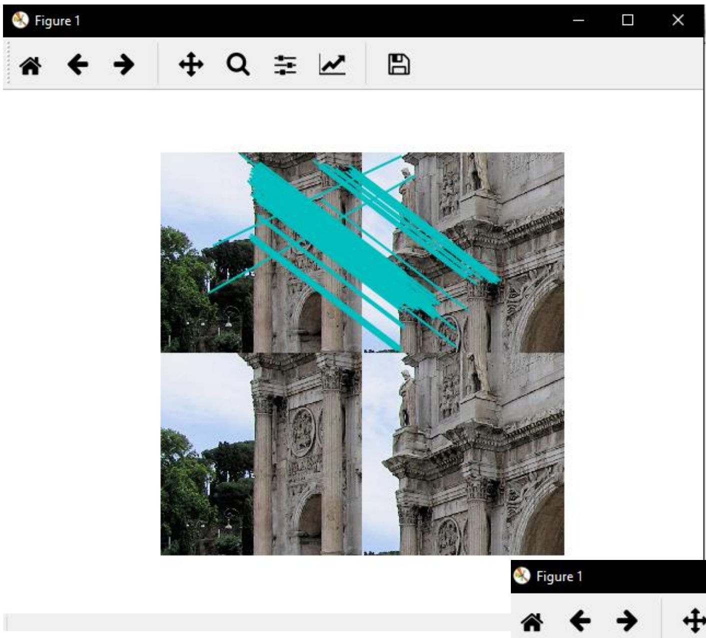
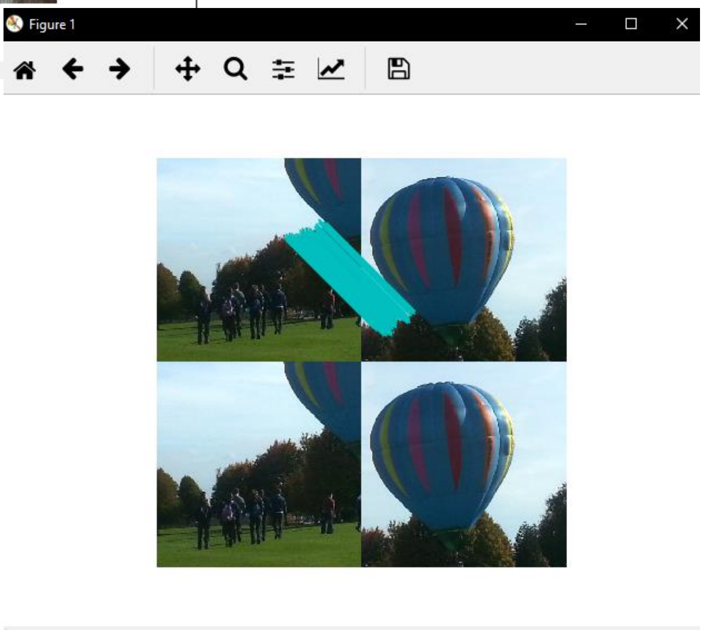


Figure 5(a) above and (b) right, showing matching points in each image pair



It is clear from these images that the program is matching the points correctly. At a 0.95 threshold, the arches image is almost all correctly matched points, with only two mismatches to be seen. The image of the balloon shows even better results, with every corner being matched correctly.

RANSAC

RANSAC is a technique for fitting to noisy data sets which contain outliers. Although this data has very few outliers, RANSAC could still be used to find the best fit. The method involves picking matching points at random and divides the set into good inliers and bad outliers. It does this for multiple random matching points until an exhaustive fit has been completed.

The goal of this method is to find the correct translation in order to stitch the two images together, i.e. to see how far along the x and y axes to move the images.

Conclusion

Although this project didn't complete the exhaustive RANSAC and stitching of images successfully, there was still a number of techniques used throughout the project which can be considered a success.

The Harris corners were detected correctly in each image and the matching corners were found in the accompanying image. The matches were then plotted and it was clear to the human eye the translation that should take place in order to stitch the images together correctly to form a larger image.