CS172 Computer Vision I:
Homework 3: Image Stitching

WU Daqian
2018533262
wudq1@shanghaitech.edu.cn

Abstract

In this homework, I focus mainly on the problem of panoramic image stitching. This homework is mainly based on the work of Matthew Brown and David G. Lowe, and is an simplified version of the implementions provided in.

1. Introduction

In this paper, I would describe an invariant feature based approach to panoramic image stitching. It will be able to automatically discover the matching relationships between provided images. Firstly, each picture will be detected and described using Scale Invariant Feature Transform[Lowe] descriptors, where key points are detected using Difference of Gaussians(DoG) and described using 128 dimension vector representing each patch[2]. The following step of making a paranoma stitch is to match the keypoints in each image in order to understand the transformation between two images. Keypoints are matched through FLANN based KD-tree, where then the key point matches are filtered to get more unique and certain pairs of matches. Thus using RANSAC algorithm to calculate the homography upon these keypoints will quite bring us the right result of transformation between images. All that is left to do is to blend the imaages together using previous implemented belnding methods[4].

2. Key points and matching

Ideally, key points in the image whould be found by scanning an image with Laplacian of Gaussian kernel of different window size. However, LoG is usually costly to run, an acceptable approximation would be Difference of Gaussians(DoG). Furthermore, use Taylor series expansion to filter the potential keypoints locations to get more accurate results. Next, give an orientation assignment to each key loction to achieve invariance to image rotation. A neighbourhood is taken around the keypoint location de- pending on the scale, and the gradient magnitude and direction is calculated in that region[2]. An orientation histogram with 36 bins covering 360 degrees is created. The highest peak in the histogram is taken and any peak above 80% of it is also considered to calculate the orientation. To represnt our key patches corresponding to key points, a 16x16 neighbourhood around the keypoint is taken, this neighbourhood is then split into 4×4 blocks of 4×4 region, for which an 8 bin orientation histogram is created. So over- all, for each patch, we have 128 bin values, this forms the 128 dimension vector representing a patch[2].

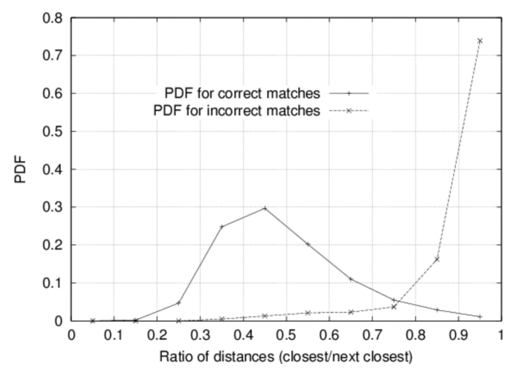

Figure 1. **Probability distribution function of mathces**

2.1. Filtering good keypoints

It is inevitable that there will be some bad matching due to repeated patterns are simply flaws of the algorithm. Therefore, we need to promarily filter out those that are not likely to be good matches As is shown in **Fig. 1**, we can define the ratio between the nearest match and the second nearest match of key points, from the distribution, we can easily see that the higher the ratio is the less reliable the match is[3]. In the homework, the filtering ratio has been set to $GOOD\_MATCH\_RATIO = 0.7$

2.2. KD-Tree

A KD-Tree is a binary search tree where data in each node is a K-Dimensional point in space. In short, it is a

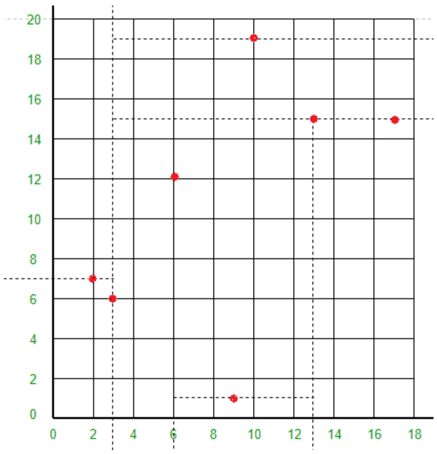

Figure 2. **Graphical demonstration of KD-Tree** Each point is a node in the tree that seperates the space or subspaces into two halves.

space partitioning data structure for organizing points in a K-Dimensional space. With this structue, we can quickly find matching pairs of keypoints[4].

2.3. Results of matching

Fig. 3 are the results of matching between image pairs, one thing to bare in mind is that before doing the mathing, all images are resized by a factor of $RESIZE\ RATIO = 8$. As there are not many intercrossing connections in **Fig. 3**, we can say that the matches are rather good and firm.

3. Calculation homographies

The main algorithm for this part is **RANSAC**.The RANSAC algorithm is an algorithm for robust fitting of models in the presence of many data outliers. There are simply four steps: i)Select sample of m points at random, ii)Calculate model parameters that fit the data in the sample, iii)Calculate error function for each data point, iv) Select data that support current hypothesis, v) Repeat sampling[5].

3.1. Direct linear transform

Sloving the homography is essentially solving (1).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{1}$$

Typically, such eqution is solved by iterations or iterative improvement. However, there is a faster and less costly way, that is by Direct Linear Transform. (1) can be turn into (2)

$$A_i h = 0 \tag{2}$$

where

$$A_i = \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & xy' & y' \end{bmatrix}$$

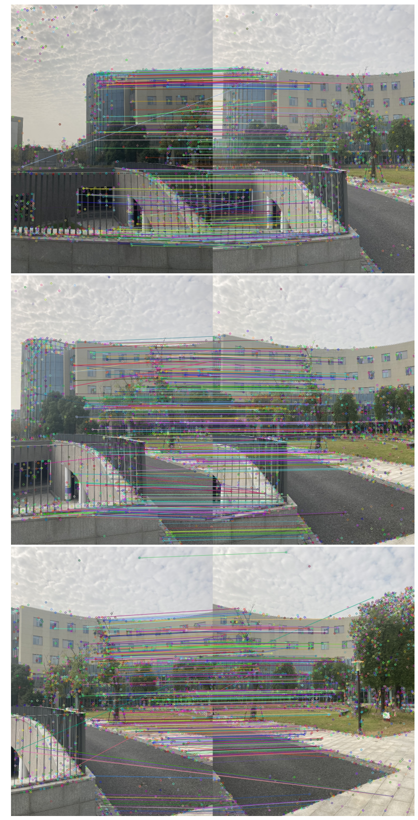

Figure 3. **Keypoint matching results** Lines connecting two dots represents that those two points are matched.

$$h = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 \end{bmatrix}^T$$

With this, we can apply singular value decomposition(SVD) to solve H. the steps are as follows:

- For each correspondence create A_i
- Concatenate into single $2n \times 9$ matrix A
- Compute SVD of $A = U \sum V^T$
- Store Eigenvector of the smallest Eigenvalue
- Reshape to get H

We use that good matches that we found previously and apply the ransac algorithm to get robust homoghies between images.

3.2. Results

Fig. 5 are some of the results of images after wraping homogrphies perspective, the first(or center) image has a identity homograhy thus is not shown.

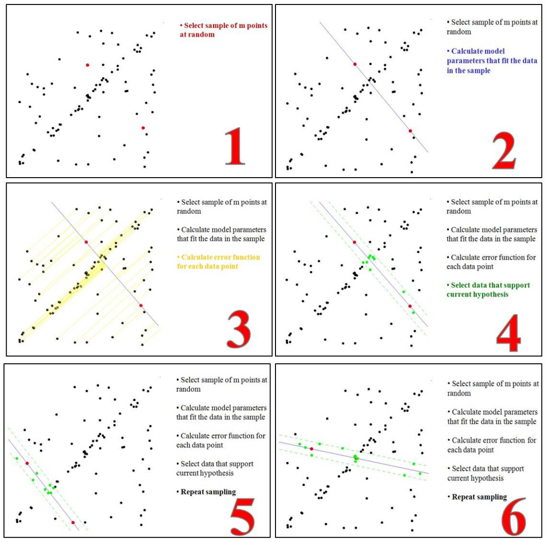

Figure 4. **RANSAC Illistrated**

4. Stitching

4.1. Pairwise

Here are the results of pairwise stitching, namely stitching the wrapped image onto the stionary image with a non-zero mask. Results are shown in **Fig. 6**.

4.2. Panaroma

Doing the whole panaroma requires propagating homographies between images. Let $H_{1,2}$ and $H_{2,3}$ be the homography between image1 image2 and image2 image3. It is possible to use image2 as the stationary image in this case, however, for more complicated cases, we might have to imagine that image1 is the stationary image. Hence, in order to stitch image3, both $H_{1,2}$ and $H_{2,3}$ has to be applied to it. Here is the result in **Fig. 7**.

5. Need of improvement

5.1. Bundle Adjustment

As seen in the results, concatenation of pairwise homographies would cause accumulated errors and disregard multiple constraints between images, this yeilds poor results. The reason tothis not because we did not condiser these images in a budle, if we using the property that they are a bundle, we may compute the geometric parameters (orientation and focal length) of each camera, namely solve for all of the camera parameters jointly[4].

5.2. Gain compensation

Bundle Adjustment solves geometric parameters, but photometric parameters, namely the overall gain between

Figure 5. Each image applied to its own homography.

imagesis also of great importance.There is still room for aplience for such an method[4].

5.3. Multi-Band Blending

Ideally, pixels on the edge of stitches should have the same intensity, but in reality, this is not the case, it is better if we could apply seamless cloning[1] to solve this matter. Even better, if we could blemd these images as a whole, using multi bands, the results will be more appealling[4].

References

[1] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[2] David G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the 7th IEEE Int. Conf. on Computer Vision*, page 11501157, 1999.

[3] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV 60 (2), pp. 91-110*, 2004.

Figure 6. Pairwise stitching with the image to the left stationary.

Figure 7. Panaroma result.

[4] David G. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 2007.

[5] R. C. Bolles M. A. Fischler. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM, Vol 24, pp 381-395*, 1981.