Exercise on Multiple Images Stiching

xiahaa@space.dtu.dk

March 29, 2019

In this exercise, you will work on using all algorithms you have developed yet to do the stiching for more than two images.

1 Key Steps

In order to stich more images, you will have several options:

1.1 Option 1

1. Choose two images and stich them.
2. Use the stiched image and a third image, stich again.
3. Iterate the previous two steps until all images have been added.

The advantage of this option is easy to implement. However, the disadvantage is the complexity will grow significantly when more images have been stiched because you will have one image quite large.

1.2 Option 2

The second option is similar to this tutorial[1].

1. load images.
2. denote previous image as $i-1$ and current image as i, detect feature points and descrptors for $i-1$ and i, match them, and compute the homography (although in the tutorial, the geometric transformation is used, we can still use homography).
3. compute the size for the final stiched image (You can use **maketform, imtransform** and the homography $\mathbf{H}$ to compute the output limits for each image). Then find the bounding box that enclose all stiched images.
4. find the center of the bounding box and then find the image which will be the nearest one to the center after transformation. Make this image as the base.
5. When you have the base image, you need recompute the homography relative to this base image. For example, after step 2, you will have $\mathbf{H}_2^1, \mathbf{H}_3^2, \cdots$. Here the superscript means to which image, subscript means from which image, so $\mathbf{H}_2^1, \cdots$ means the homography from image 2 to image 1. Now you set image 3 as the base, then the homography from image 1 to image 3 can be computed as $\mathbf{H}_1^3 = (\mathbf{H}_3^2)^{-1}(\mathbf{H}_2^1)^{-1}$ since $\mathbf{H}_1^3 = \mathbf{H}_2^3\mathbf{H}_1^2$ and $\mathbf{H}_1^2 = (\mathbf{H}_2^1)^{-1}$, $\mathbf{H}_2^3 = (\mathbf{H}_3^2)^{-1}$[2].
6. The final step is to warp each image to the stiched image and blend.

[1] https://www.mathworks.com/help/vision/examples/feature-based-panoramic-image-stitching.html

[2] If $\mathbf{H}$ is from i to j, the $\mathbf{H}^{-1}$ is from j to i.

1.3 Option 3

In my code **main_stich_mutiple.m**, I am using a different strategy as follows

1. load every image, do feature detection and description;
2. do matching for each pair, there will be C_n^2 pairs. You will get a matrix of $n \times n$ with each element denotes how many matches obtained. For example, the $(2, 3)$ element contains the number of features matched for image 2 and image 3. Note this matrix is symmetric which means $(i, j) == (j, i)$.
3. Next, I will treat the $n \times n$ matrix as a graph and find the minimum spanning tree which will connect all nodes with the minimum cost.
4. From that tree, a node will be selected as the root. For others, the shortest path from root node to current node will be found. This gives us the relationship from base to current image.
5. Compute homography using the corresponding shortest path.
6. Stich and blend.

The strategy will try to stich images with the order that the feature macthes can reach the maximum. For example, image 1 and image 2 have 50 matches, image 1 and image 3 have 100 matches, image 2 and image 3 have 300 matches. Then this method will pick image 2 as the base and use $3 \rightarrow 2$, $1 \rightarrow 3 \rightarrow 2$ as the order to compute new homographies.

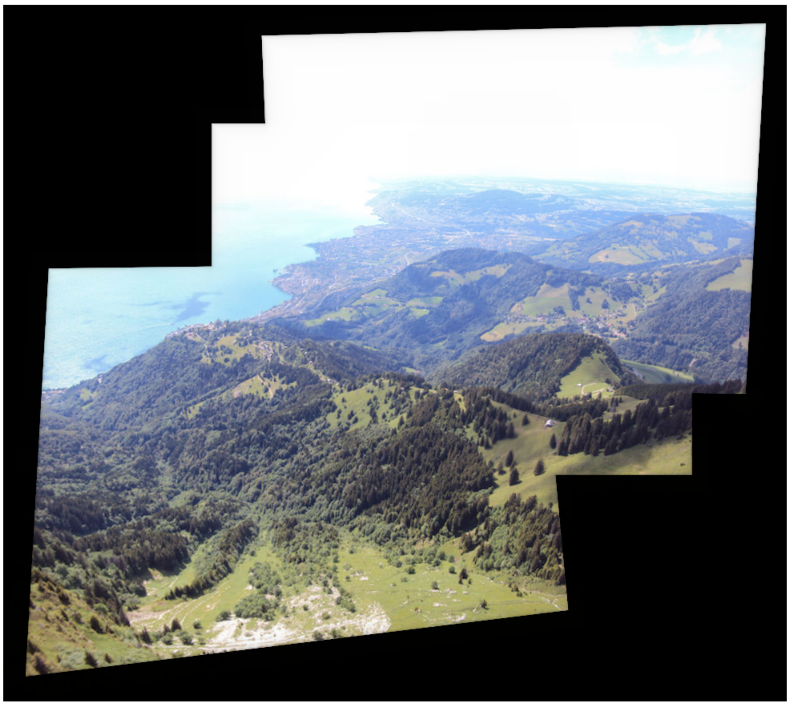

Figure 1: Example of image stich result.