README

All the image results are saved under ‘RESULT’.

Origin images and scripts are saved under current directory.

Images named ‘firelink_center’, ‘firelink_left’ and ‘firelink_right’ are used for challenge 1f.

Here is a brief introduction of each script. Detailed notes are commented in each script.

‘applyHomograph’

Apply the homograohy, compute the corresponding destination points of source points according to the homograohy. Note that the result need to be rescaled so that it will have the right value.

‘backwardWarpImg’

Using backward method, map the pixels of destination canvas back to the source image. And return the mapped image and mask.

‘blendImagePair’

In ‘overlay’, simply draw the destination image over the source image.

In ‘blend’, first reverse the mask to have the image has value 0 and the boundary has value 1. Then using ‘bwdist’ to compute the distance of each point to the center and use the result as the weight. And use the weight to blend the images.

‘conputeHomography’

Use the method given in lecture, to compute the eigenvector corresponding to the minimum eigenvalue of $A^T A$ and that is the homography we desire.

‘runRANSAC’

Implementing the RANSAC algorithm. Randomly choose four pair of matched points, compute the homography and find the number of pairs of points which fit to (here ‘fit to’ means error is less than the given epsilon) this homography. Loop N times where N is given as an input. Return the homography with most matched pairs and the indexes of these pairs.

‘showCorrespondence’

Draw two images side by side, and draw a line through each matched pair of points.

‘stitchingImg’

Using iterations to stitch images. In each iteration:

First find the matched pairs of points using SIFT.

Using ‘runRANSAC’ to find the most likely homography of these two images.

Create an extended image with the size of stitched image and draw the current image on it. Using ‘backwardWarpImg’ to generate a mapped image with a same size.

Using ‘blendImagePair’ to blend current image and new image together and choose the result as current image for next iteration.