EE5178 : Modern Computer Vision

Programming Assignment 4: Panoramic Stitching

Instructions

1. Please use moodle discussion threads for posting your doubts.
2. Before posting any question, check if the same question has been asked earlier.
3. Submit a single zip file in the moodle named as `PA4_Rollno.zip` containing the report and folders containing corresponding codes.
4. Read the problem fully to understand the whole procedure.
5. Comment your code generously.
6. Put titles to all of the figures shown.
7. Save your `iPython` notebook as an `html` file with all the figures and results and submit it along with the iPython notebook containing your code.

Panoramic Stitching

In this problem, we will develop an algorithm for stitching a panorama from overlapping photos (Figure 1), which amounts to estimating a transformation that aligns one image to another. To do this, we will compute SURF/SIFT/ORB features in both images and match them to obtain correspondences. We will then estimate a homography from these correspondences, and we'll use it to stitch the two images together in a common coordinate system. In order to get an accurate transformation, we will need many accurate feature matches. Unfortunately, feature matching is a noisy process: even if two image patches (and their SURF/SIFT/ORB descriptors) look alike, they may not be an actual match. To make our algorithm robust to matching errors, we will use RANSAC, a method for estimating a parametric model from noisy observations. We will use the obtained homography transformation to do panoramic stitching. We have provided you with two input images and also a starter code.

(a) Image pair (b) Stitched panorama

Figure 1: Panorama produced using our implementation.

Tasks

1. Implement `get_features(img)` to compute SURF/SIFT/ORB features for both of the given images. Implement `match_keypoints(desc1, desc2)` to compute key-point correspondences between the two source images using the ratio test. Run the plotting code to visualize the detected features and resulting correspondences.
   Hint: *You can use existing libraries.*

2. Write a function `find_homography(pts1, pts2)` that takes in two $N \times 2$ matrices with the x and y coordinates of matching 2D points in the two images and computes the 3×3 homography H that maps `pts1` to `pts2`. You can implement this function using direct linear transform (DLT). Report the homography matrix.
   Note: *You should implement this function on your own.*

3. Your homography-fitting function from (2) will only work well if there are no mismatched features. To make it more robust, implement a function `transform_ransac(pts1, pts2)` that fits a homography using RANSAC. Run the plotting code to visualize the point correspondences after applying RANSAC.
   Note: *You should implement this function on your own.*

4. Write a function `panoramic_stitching(img1, img2)` that produces a panorama from a pair of overlapping images using your functions from the previous parts. Run the algorithm on the two images provided. Report the panorama stitched image.
   Note: *You should implement this function on your own. Use inverse mapping while stitching.*

5. Extend the algorithm to handle $n = 3$ images, use these given images. We have also provided a reference panoramic stitched image to compare your result with. Report all visualizations as mentioned for $n = 2$ case, along with two homography matrices with respect to the center image.
   Note: *You should implement this function on your own. Take the center image as the reference and find the homography transformations with respect to this center image, which doesn't need further transformations and also produces the most aesthetically pleasing panorama.*

–end–