Uniwersytet Wrocławski | Instytut Informatyk

Computer Vision and
Photogrammetry
Winter 2016-17

LAB 6: ROBUST ESTIMATION

DUE: 24 DECEMBER.

Task 1 (50) RANSAC computation of E.

Use your code from Labs 4 and 5 to compute matches between 2 images. Select a pair of images and compute the essential matrix using these features. Because not all matches are correct, you need to use RANSAC to find a set of good correspondences (inliers). You can use the code from Lab 2 to compute an estimation of E from 8 points. To compute the inliers use the distance of the point to the epipolar line. Take into account how many iterations you need to run to guarantee that you got a good E.

Use the following K matrix.

$$\begin{bmatrix} 2759.48 & 0.00000 & 1520.69 \\ 0.00000 & 2764.16 & 1006.81 \\ 0.00000 & 0.00000 & 1.00000 \end{bmatrix}$$

If your keypoint detector, descriptor, or matching are not working properly, you can use some external library for this, for example Opencv.

Task 2 (50) Adding Images.

Find P1 and P2, Lab 2 and triangulate the matches (this was also optional in Lab 2). Now, with the matches between images 1 and 2 and a 3rd image, use the 3d points to calculate the camera matrix of P3, as we did in Lab 1. Again your matches might be incorrect, so you need to use a RANSAC. In this case, you use the reprojection error to identify outliers.

Once you have P3, you might be able to add some 3D points from novel matches, using the same method as before.

Add sequentially the rest of the images.

Save your results in a file, containing the camera matrices, the 3d points and the 2D correspondences. Something similar to NVM (see notes)

Save the 3D point cloud as a PLY file. Add colour from the original images (average of the available 2D correspondences.)

Extra: Projective Reconstruction and Auto-calibration

Do the same as in task 1, but compute the Fundamental Matrix F instead. Pick the canonical camera matrices, and continue with task 2. Your reconstruction now is a projective one, so you need to do auto-calibration to convert it to a metric one. You can use one of the methods described in http://www.cs.unc.edu/~marc/tutorial/node90.html. We now that the pictures were taken with the same camera, so http://www.cs.unc.edu/~marc/tutorial/node92.html is valid.

Notes.

I have included a NVM file computed with VisualSFM. Here the format. http://ccwu.me/vsfm/doc.html#nvm There you have a lot of reconstructed points, and their 2D correspondences. Also camera positions and orientations using the format described in the link above. Cameras are also in files *.P (for the camera matrices) and *.camera for the camera parameters. Your parameters won't match, but should be consistent, up to a scale, and a translation.

You shouldn't use these points for the RANSAC computation, but you can use them for testing, debugging, etc.

You have other similar datasets here.

http://cvlabwww.epfl.ch/data/multiview/denseMVS.html