Homework 3: Geometric vision

This assignment has two parts:

1. Fundamental matrix estimation and triangulation
2. Affine factorization

Part 1 Epipolar geometry estimation, 3D reconstruction

The goals of this part of the assignment are fundamental matrix estimation and triangulation. You will be using these two image pairs:

HW3_part1_data.zip contains full size images, matching points, camera matrices, and sample code.

Directions

1. Load the image pair and matching points file into MATLAB (see sample code in the data file).

2. Fit a **fundamental matrix** to the matching points. Use the sample code provided to visualize the results. Implement both the **normalized** and the **unnormalized** algorithms. In each case, report your residual, or the mean squared distance *in pixels* between points in both images and the corresponding epipolar lines.

3. Load the camera matrices for the two images (they are stored as 3x4 matrices and can be loaded with the load command, i.e., P1 = load('house1_camera.txt'); Find the centers of the two cameras. Use linear least squares to **triangulate** the position of each matching pair of points given the two cameras. Display the two camera centers and reconstructed points in 3D. Also report the residuals between the observed 2D points and the projected 3D points in the two images.

Tips and Details

- For fundamental matrix estimation, don't forget to enforce the rank-2 constraint. This can be done by taking the SVD of F, setting the smallest singular value to zero, and recomputing F.
- Lecture slide shows two slightly different linear least squares setups for estimating the fundamental matrix (one involves a homogeneous system and one involves a non-homogeneous system). You may want to compare the two and determine which one is better in terms of numerics.
- Recall that the camera centers are given by the null spaces of the matrices. They can be found by taking the SVD of the camera matrix and taking the last column of V.
- For triangulation with linear least squares, it is not necessary to use data normalization (in my implementation, normalization made very little difference for this part).
- Plotting in 3D can be done using the plot3 command. Use the axis equal option to avoid automatic nonuniform scaling of the 3D space. To show the structure clearly, you may want to include snapshots from several viewpoints. In the past, some students have also been able to produce animated GIF's, and those have worked really well.

Extra Credit

- Use your putative match generation and RANSAC code from Homework 2 to estimate fundamental matrices without ground-truth matches. Compare the quality of the result with the one you get from ground-truth matches.

Part 2: Affine Factorization

The goal of this part of the assignment is to implement the Tomasi and Kanade affine structure from motion method as described in this lecture. You will be working with Carlo Tomasi's 101-frame hotel sequence:

HW_part2_data.zip file, including all the 101 images and a measurement matrix consisting of 215 points visible in each of the 101 frames (see readme file inside archive for details).

Part 2 Directions

- Load the data matrix and normalize the point coordinates by translating them to the mean of the points in each view (see lecture for details).
- Apply SVD to the 2M x N data matrix to express it as D = U * W * V' where U is a 2Mx3 matrix, W is a 3x3 matrix of the top three singular values, and V is a Nx3 matrix. Derive structure and motion matrices from the SVD as explained in the lecture.
- Use plot3 to display the 3D structure (in your report, you may want to include snapshots from several viewpoints to show the structure clearly). Discuss whether or not the reconstruction has an ambiguity.
- Display three frames with both the observed feature points and the estimated projected 3D points overlayed. Report your total residual (sum of squared Euclidean distances, in pixels, between the observed and the reprojected features) over all the frames, and plot the per-frame residual as a function of the frame number.

Part 2 Extra Credit

- Attempt to eliminate the affine ambiguity using the method described in the lecture.
- Create a textured 3D model of the reconstructed points. For example, you can compute a Delaunay triangulation of the points in 2D, then lift that mesh structure to 3D, and texture it using the images.

Instructions for turning in the assignment

Submit your PDF report and zipped code archive, named **studentnumber_hw3.pdf** and **studentnumber_hw3.zip** via course web.