SGN-45006 Fundamentals of Robot Vision
Exercise Round 7

April 15, 2019

For these exercises you will need Python or Matlab which should be available on the university computers. Return your answers as a pdf along with your modified code to Moodle. Exercise points will be granted after a teaching assistant has checked your answers. Returns done before the solution session will result in maximum of 3 points, whereas returns after the session will result in maximum of 1 point.

Task 1. Fundamental matrix estimation. (Programming exercise) (1 point)

See the comments in `Fmatrix_example` and implement the two missing functions:

a) Implement the eight-point algorithm in `estimateF`. The algorithm is described below and in the lecture slides.

Let's denote $\mathbf{x} = (u, v, 1)^\top$ and $\mathbf{x}' = (u', v', 1)^\top$.

The eight-point algorithm can be implemented by solving the following homogeneous linear system:

$$(u'u, u'v, u', v'u, v'v, v', u, v, 1) \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix} = 0$$

This solution is then enforced rank-2 constraint by taking SVD and then reconstructing using only the two largest singular values.

b) Implement the missing denormalization used in normalized eight-point algorithm `estimateFnorm`. If $\mathbf{T}$ and $\mathbf{T}'$ are the normalizing transformations in the two images, the fundamental matrix in original coordinates is $\mathbf{T}'^\top \mathbf{F} \mathbf{T}$

The epipolar lines obtained with both F-matrix estimates should be close to those visualized by the example script. **Include an image with the epipolar lines obtained with both F-matrix estimates, as well as your version of the code.**

Task 2. Two-view structure from motion. (Programming exercise) (2 points)

In this exercise you will estimate the fundamental matrix for a pair of uncalibrated images and recover a pair of camera projection matrices that are compatible with the estimated fundamental matrix. Thereafter, triangulation of point correspondences using the aforementioned projection matrices gives a projective reconstruction of the scene, which is visualized in the example script.

Run the script `two_view_structure_from_motion_example` and proceed as follows (do the tasks and answer the questions):

a) The first part of the code calibrates the cameras using known dimensions of the shelf and visualizes a wireframe model of the shelf projected onto the images. How are the cameras calibrated here?

b) The second part triangulates and visualizes a projective reconstruction of the wireframe model. Give an explanation why the model looks distorted but is anyway correct. (See discussion in the lecture slides.)

c) In the third part you should project the distorted wireframe model onto the two images and check that it matches the outlines of the book shelf. **Include a picture that illustrates the projected model, as well as your version of the code.**

d) Finally, describe what kind of information could be used to upgrade the projective reconstruction to a similarity reconstruction, where angles and ratios of lengths are the same as in the real one, without knowing the dimensions of the book shelf. (Hint: See lecture slides.)