Photogrammetric Computer Vision
Final Project

Winter Semester 24/25

Submission Deadline: 23.03.25 23:00 pm

Essential Matrix Estimation and Non-linear Optimization

Task 1 – Essential Matrix Estimation

While the *fundamental matrix* represents the relative orientation of an image pair for the uncalibrated case (i.e. without information about the interior orientations resp. camera intrinsics), the *essential matrix* covers the calibrated case. Using algebraic projective geometry, the *epipolar geometry* of a calibrated camera pair can, thus, be expressed by the *essential matrix* $\mathbf{E}$.

a) Use the data in the `calib_points.dat` file, which provides 12 *homologous image coordinates* $\mathbf{x}_1 \leftrightarrow \mathbf{x}_2$, as well as *corresponding 3D object points* in the format (x_1, y_1, x_2, y_2, X, Y, Z). Use this information to compute the *calibration matrices* $\mathbf{K_1}$ and $\mathbf{K_2}$ for each of the cameras applying a method from the lectures. (For a detailed example of reading a file in MATLAB, see the description of *exercise 5.*)

b) Based on the computed *calibration matrices* $\mathbf{K_1}$ and $\mathbf{K_2}$ estimate the *essential matrix* $\mathbf{E}$, which relates the two camera views.

c) Resolve the *fourfold ambiguity* of the essential matrix by selecting the geometrically plausible solution.

d) Compute the *epipolar lines* from the essential matrix. Finally, visualize the epopolar lines from the two camera views and the corresponding image points.

Task 2 – Non-linear Optimization

Singular value decomposition (SVD) provides an optimal solution with respect to the algebraic error. The goal, however, is to obtain a solution optimal with respect to the *geometric error*. The algebraic solution can serve as a starting point for further non-linear optimization.

a) Compute the *geometric error* based on the solution from Task 1.

b) Perform a non-linear optimization by means of the *indirect optimization* using a built-in function of your choice (Levenberg-Marquart, etc.).

c) Re-calculate the geometric error. Report and comment on the results.

Remark: Your final submission for the project should include the **source code** of your MATLAB (Octave) implementation and a PDF document with **short documentation** (maximum 5 pages). Any external sources that you use should be included in the reference section of your documentation.

Computer Vision in Engineering – Prof. Dr. Rodehorst
M.Sc. Mariya Kaisheva
mariya.kaisheva@uni-weimar.de