Exercise 4

Matthias Hoffmann, Room 10.136 matthias.hoffmann@cs.fau.de
Yan Xia, Room 09.157 yan.xia@cs.fau.de

Normalized Eight Point Algorithm, Fundamental Matrix, and Epipolar Geometry

1 Get Point Correspondences of 2 images

Use the programm **get_point_correspondences.m** to get 10 corresponding points (we can change the number of corresponding points). We select the points alternating in the left and right image. The program saves the correspondences in 2 variables: left_image_points and right_image_points. We need the correspondences to build up the measurement matrix A and to compute the fundamental matrix F afterwards.

Figure 1: Left and right test image.

2 Normalized Eight Point Algorithm for Fundamental Matrix F

An overview of the algorithmic steps is listed in the box. x_i are the image coordinates in the left image, x'_i the coordinates in the right image respectively.

Objective

Given $n \geq 8$ image point correspondences $\{x_i \leftrightarrow x'_i\}$, determine the fundamental matrix F such that ${x'_i}^T F x_i = 0$.

Algorithm

1. **Normalization:** Transform the image coordinates according to $\hat{x}_i = T x_i$ and $\hat{x}'_i = T' x'_i$, where T and T' are normalizing transformations consisting of a translation and scaling.
2. Find the fundamental matrix $\hat{F}'$ corresponding to the matches $x_i \leftrightarrow x'_i$ by
   (a) **Linear solution:** Determine $\hat{F}$ from the singular vector corresponding to the smallest singular value of $\hat{A}$, where $\hat{A}$ is composed from the matches $\{x_i \leftrightarrow x'_i\}$.
   (b) **Constraint enforcement:** Replace $\hat{F}$ by $\hat{F}'$ such that $\det \hat{F}' = 0$ using the SVD.
3. **Denormalization:** Set $F = {T'}^T \hat{F}' T$. Matrix F is the fundamental matrix corresponding to the original data $\{x_i \leftrightarrow x'_i\}$.

(Multiple View Geometry, R. Hartley and A. Zisserman, p. 282)

2.1 Normalization (balancing) and denormalization

Image coordinates are sometimes given with the origin at the top-left of the image (all image coordinates are positive) or with the origin at the center. The question immediately occurs whether this makes a difference to the results of computing the transformation F. As a first step of normalization, the coordinates in each image are translated (by a translation for each image) so as to bring the centroid of the set of all points to the origin. The coordinates are also scaled so that on the average a point **p** is of the form $p = (x, y, w)^T$, with each of x, y and w(homogeneous coordinate) having the same average magnitude. An isotropic scaling factor is chosen so that the x-/y-coordinates of a point are scaled equally. Implement this balancing of the input points as an option, which you can switch on/off. (Multiple View Geometry, R. Hartley and A. Zisserman, p. 107)

The coordinates in each image are translated to bring the centroid of the set of all points to the origin independently. Afterwards the coordinates are scaled so that the average distance of a point **p** from the origin is equal to $\sqrt{2}$.
The following steps have to be done for each frame:

• Translation:

$$t_x = \frac{1}{N} \sum_{k=1}^{N} p^i_{x,k}, \qquad t_y = \frac{1}{N} \sum_{k=1}^{N} p^i_{y,k}, \tag{1}$$

where $t = (t_x, t_y)$ is the centroid or 'center of mass' of the $k = 1, ..., N$ points p^i_k in the according frame.
Now we can center the points by:

$$mc = p_k^i - t. \quad (2)$$

• Scaling: The distance of each new calculated point to the origin (0,0) is defined by:

$$dc = \sqrt{\sum (mc)^2}. \quad (3)$$

dc divided by the number of tracked points, results in the average distance to the origin. To satisfy the criteria that the average distance of a point $\mathbf{p}$ from the origin is equal to $\sqrt{2}$, we have to apply a scale factor s.
The normalized data can be computed by applying the transformation matrix
$T = \begin{pmatrix} s & 0 & -st_x \\ 0 & s & -st_y \\ 0 & 0 & 1 \end{pmatrix}$. We have to adjust the result to undo the effect of the normalization.
This can be done by: $F = T'^T \hat{F}' T$. T' is the transformation of the right image, T of the left image.

2.2 Computing the Fundamental Matrix

First, we have to set up the measurement matrix A (see slides: "Magnetic navigation" 18/39): $Af = 0$. The minimum case for setting up A is seven point correspondences. This leads to a 7×9 matrix A with rank 7. The solution to the equations $Af = 0$ is a 2-D space of the form $\alpha F_1 + (1 - \alpha) F_2$, where α is a scalar variable (Multiple View Geometry, R. Hartley and A. Zisserman, p. 281).
If we use 8 point correspondences, we get a 8×9 matrix A with rank 8. Then it is possible to solve for f up to scale.
Pixel coordinates have to be converted into homogeneous coordinates ($w = 1$). The nullspace of the matrix A is the solution for f. We have to reshape the vector to get a 3×3 matrix $\hat{F}$. We have to enforce the rank 2 criterion $\rightarrow$ fundamental matrix $\hat{F}'$. To get the final fundamental matrix F, we have to apply the transformations of the balancing step 2.1.

3 Computing the epipolar line

F maps a point to its epipolar line in pixel coordinates by: $I^T = f x_i$. x_i is the homogeneous coordinate of the left image. I^T is the point in the right image. We have to compute the epipolar line in the right image by the line equation: $ax + by + c = 0$. Additionally, we compute the epipolar line in the left image. We know that the nullspace of F is the right epipole and the nullspace of F^T is the left one ($\widehat{=}$ third column of U: right epipole; third column of V: left epipole).
For our 2 test images, both epipoles lie outside of the visible image.

4 References

This exercise treats the Eight Point Algorithm as well as the normalization steps that improve its numerical stability.

For details, please read the following papers:

1. In Defense of the Eight-Point Algorithm
   R. Hartley
   http://ieeexplore.ieee.org/iel1/34/13258/00601246.pdf

2. Revisiting Hartley's Normalized Eight-Point algorithm
   W. Chojnacki, M. Brooks, A. van den Hengel, D. Gawley
   http://ieeexplore.ieee.org/iel5/34/27551/01227992.pdf?arnumber=1227992

For general information in the topics epipolar geometry and eight point algorithm, we refer to:
Multiple View Geometry in computer vision
R. Hartley, A. Zisserman
Cambridge University Press
Part II "Two-View Geometry", Chapters 9–14

5 Useful MATLAB commands

svd, reshape, eye, save, load, plot

We will work interactively on the implementation of the above mentioned tasks!