Assignment 2

Jenni Simon
09-116-005

Photometric Stereo

1. Epipolar lines estimation The first part of this assignment is concerned with estimating the fundamental matrix $\mathbf{F}$ from user-specified corresponding points in two images of the Zytglogge in Bern (See figure 1) using the eight-point algorithm.

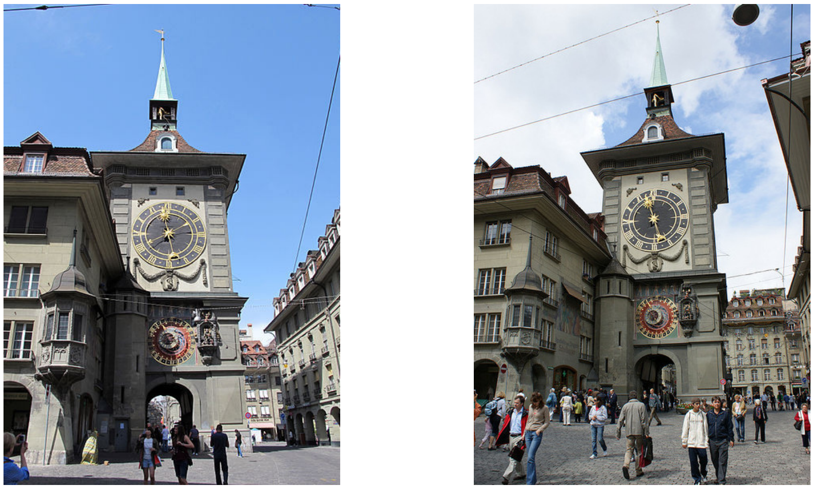

Fig. 1: Input images of the Zytglogge taken from different positions.

Gathering corresponding points In order to estimate $\mathbf{F}$, a minimum of eight corresponding points in both images are necessary. The user is therefore asked to input eight correspondences between the left and right images displayed to him. For every correspondence the user is asked to first click the point in the left image and then click the corresponding point in the right image. The chosen points are marked such that points belonging to a pair have the same color and different pairs have different color (figure 2). It should be noted the the eight points should not be co-planar for the algorithm to work properly.

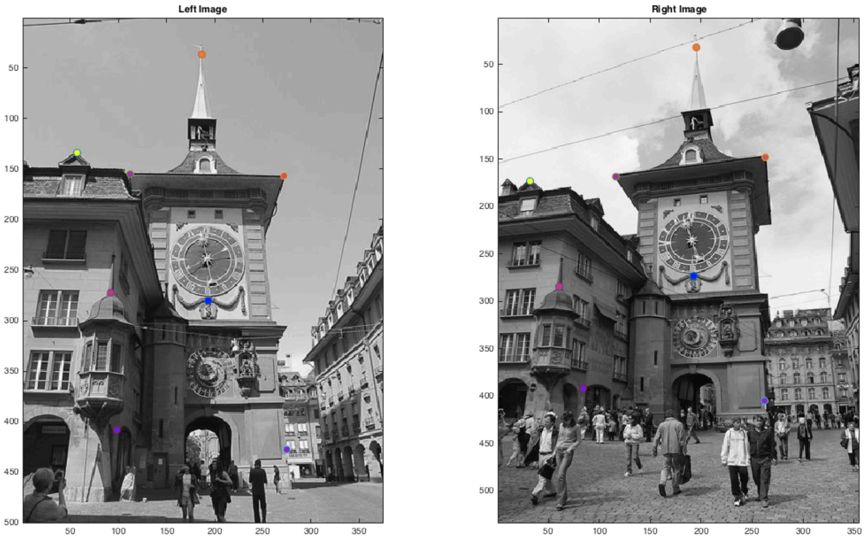

Fig. 2: Correspondances specified by the user.

Algorithm Let x_{i1} and x_{i2} be homogeneous image-coordinates $x = (u, v, 1)$ of the i^{th} matched pair in the left and right image respectively. The fundamental matrix $\mathbf{F}$ satisfies the condition

$$x_{i1}^T \mathbf{F} x_{i2} = 0 \tag{1}$$

If we set $f^T = [F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33}]$ as the vectorisation of $\mathbf{F}$, we can express the condition as inner product:

$$[u_{i2}u_{i1}, u_{i2}v_{i1}, u_{i2}, v_{i2}u_{i1}, v_{i2}v_{i1}, v_{i2}, u_{i1}, v_{i1}, 1] \cdot f = 0 \tag{2}$$

Given N matching points we can further stack them in an $N \times 9$ matrix A and arrive at the linear system

$$A \cdot f = 0 \tag{3}$$

As the solutions are ambiguous up to scale, solving for f while avoiding the trivial solution $f = 0$ is a minimum direction problem (constrain $|f| = 1$) and can be solved by looking at the SVD of $A = USV^T$. The vector f is then given by the last column of V. In general, the resulting fundamental matrix $\mathbf{F}$ (obtained by reshaping f) will not be of rank two. To enforce the singularity constraint on $\mathbf{F}$ we therefore take the SVD of $\mathbf{F}$, set the last singular value in S to zero and multiply the factors out again to get an $\mathbf{F}$ of rank two. To improve the quality of the solution, the data should be normalised prior to the computations. The reason for this is that some of the entries of A can be on the order of 1 while others can be on the order of $w \cdot h$, where w and h are the dimensions of the input images. We therefore transform the points in image 1 and 2 with transformation-matrices T_1, and T_2, such that the transformed points $x'_{i1} = T_1 x_{i1}$ and $x'_{i2} = T_1 x_{i2}$ have a mean value of zero and their mean distance to the origin is $\sqrt{2}$.

Results The following results were obtained using 8 matching pairs. $\mathbf{F}^8$ is the estimated fundamental matrix and e^8_{left}, e^8_{right} are the estimated epipoles in the left and right image respectively (epipoles were divided by homogenous cordinate):

$$\mathbf{F^8} = \begin{bmatrix} 0.0 & 0.0 & -0.008 \\ 0.0 & 0.0 & -0.008 \\ 0.005 & 0.008 & 0.426 \end{bmatrix} \quad \mathbf{e^8_{left}} = \begin{bmatrix} -715 \\ 430 \\ 1 \end{bmatrix} \quad \mathbf{e^8_{right}} = \begin{bmatrix} -369 \\ 392 \\ 1 \end{bmatrix}$$

The epipoles were computed as $\mathbf{e_{left}} = \mathbf{null}(\mathbf{F})$ and $\mathbf{e_{right}} = \mathbf{null}(\mathbf{F}^T)$. Looking at the coordinates of the epipoles, we observe an obvious error in the estimate. While the result for the right epipole is more or less plausible, the left epipole can clearly not be correct, as the x-coordinate would have to be positive. A possible explanation for this is that the algorithm is sensitive to noise in the matchings, especially when there are only few matches. In order to test this suspicion and the robustness of the algorithm, I repeated the experiment with 16 matches:

$$\mathbf{F^{16}} = \begin{bmatrix} 0.0 & 0.0 & 0.003 \\ 0.0 & 0.0 & 0.007 \\ -0.001 & -0.007 & -0.321 \end{bmatrix} \quad \mathbf{e^{16}_{left}} = \begin{bmatrix} 7514 \\ -974 \\ 1 \end{bmatrix} \quad \mathbf{e^{16}_{right}} = \begin{bmatrix} -1141 \\ 577 \\ 1 \end{bmatrix}$$

The fundamental matrix obtained in this case is quite different and the epipoles more plausible as well. However, the homogeneous coordinates in all cases were very small (likely due to the images being taken from relatively similar positions), indicating another possible explanation for the wrong epipoles. Experiments with even more matches showed only small changes in the fundamental matrix while still sometimes showing inplausible epipoles.

Figure 3 shows epipolar lines in the right image corresponding to points indicated in the left image. The fundamental matrix in this case was estimated using the eight points from figure 2.

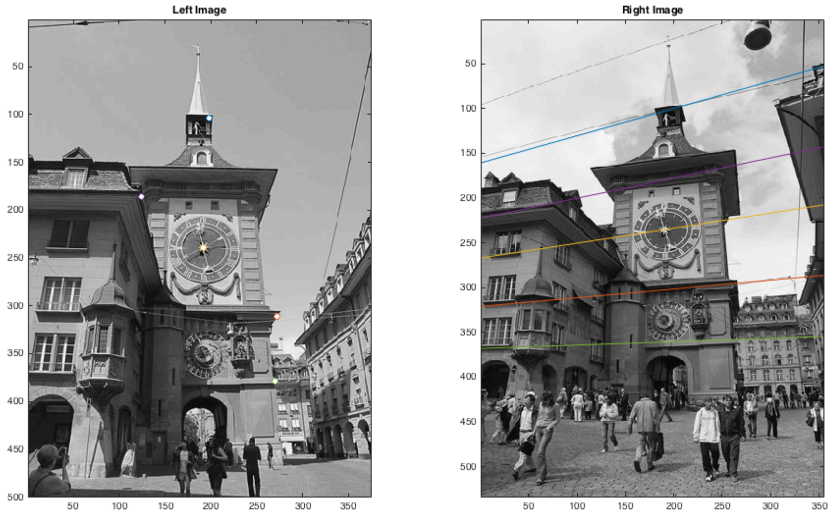

Fig. 3: Epipolar lines in the right image corresponding to the image-points in the left image.

2. Model reconstruction The second part of the assignment is concerned with estimating the essential matrix, the rotation and translation of the to views and finally reconstructing 3D-points from dense correspondences between two images of a cow.

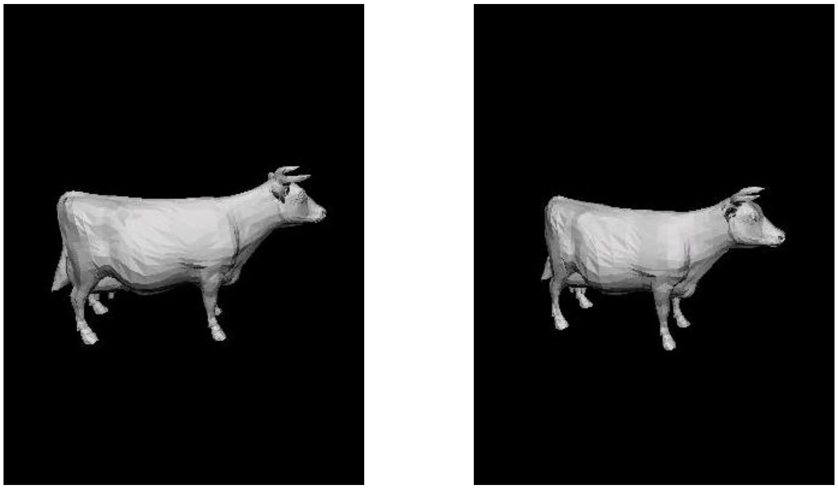

Fig. 4: Synthetic input images for the cow.

Estimating the essential-matrix The essential matrix $\mathbf{E}$ can be computed from the fundamental matrix and the intrinsic matrices Λ_1 and Λ_2 of the left and right cameras via:

$$\mathbf{E} = \Lambda_2^T \mathbf{F} \Lambda_1 \tag{4}$$

In our case $\Lambda_1 = \Lambda_2 = \Lambda$. We were given the true intrinsic parameters in the form of a matrix K and the focal length $f = f_r = f_l$ of the camera. The relation between K and Λ is given by:

$$K = \begin{bmatrix} m_x & \gamma & u_0 \\ 0 & m_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \Lambda = \begin{bmatrix} f \cdot m_x & \gamma & u_0 \\ 0 & f \cdot m_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

We can therefore compute the fundamental matrix using the eight-point algorithm and then use equation 4 to compute the essential matrix. The resulting estimate is:

$$\mathbf{E} = \begin{bmatrix} -0.008 & -2.144 & -0.881 \\ -0.04 & 0.301 & -6.283 \\ 0.005 & 5.936 & 0 \end{bmatrix}$$

Estimating the rotation and translation Given the essential matrix we are able to recover the rotation and translation of one camera to the other by decomposing the essential matrix $\mathbf{E}$. We will set the rotation matrix of the left camera to the identity and the translation vector to zero. To estimate the transformations for the right camera, let $\mathbf{E} = USV^T$ be the SVD of $\mathbf{E}$ and let W be given as:

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

We can then compute possible rotations and translations as:

$$R_1 = UW^{-1}V^T \qquad R_2 = UWV^T \tag{5}$$

$$T_\times = USWU^T \qquad t_2 = -t_1 \tag{6}$$

Where $T_\times$ and t_1 is given by:

$$T_\times = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \qquad t_1 = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

From all the possible combinations of rotations and translations, we can find the correct one by inferring the 3d-position of a matched pair (as described in the next section) and checking in which of the four possible combinations the projected 3d-position can be seen by both cameras. It should be noted however that we can not recover the scale of the translation vector and therefore constrain it to be unit length.

The ground-truth rotation and translations provided with the synthetic input images allows us to verify our results. In order to compare them however we have to translate them to a common frame of reference. We choose the left camera's orientation as frame of reference. Let R_r and R_l be the ground-truth right and left rotations and t_r, t_l be the translations. The ground truth rotations R' and translation t' of the right camera relative to the left camera is then given by:

$$R' = R_r^{-1} R_l = \begin{bmatrix} 0.929 & 0.138 & -0.345 \\ -0.129 & 0.99 & 0.048 \\ 0.348 & 0 & 0.937 \end{bmatrix}; \quad t' = R_r^{-1}(t_r - t_l) = \begin{bmatrix} 1.857 \\ -0.259 \\ 0.696 \end{bmatrix}$$

The estimated resulting from our method is given by (scaling t to the same length for better comparison):

$$R = \begin{bmatrix} 0.928 & 0.138 & -0.345 \\ -0.129 & 0.99 & 0.049 \\ 0.349 & 0 & 0.937 \end{bmatrix}; \qquad t = \begin{bmatrix} 1.861 \\ -0.261 \\ 0.685 \end{bmatrix}$$

A comparison of the results shows that we were able to accurately recover the rotation and translation with the described method.

Reconstructing 3D Having recovered the transformations between the camera positions we can now recover 3d-positions for all the matches (up to scale). Let $\mathbf{P_i} = [R_i, t_i]$ be the projection matrix for image i. The image coordinates $q_i = (x_i, y_i, 1)$ and 3d-positions $p = (u, v, w, 1)$ are related by

$$\lambda_i q_i = \Lambda_i \mathbf{P_i} p \tag{7}$$

for some λ_i. By pre-multiplying both sides with the inverse of the intrinsic matrix Λ_i and writing $q_i' = \Lambda_i^{-1} q_i$ we get:

$$\lambda_i q_i' = \mathbf{P_i} p \tag{8}$$

We can eliminate the unknown λ_i by expressing it in terms of $\mathbf{P_i}$ and p as $\lambda_i = \mathbf{P_i}(3, :) \cdot p$. Substituting λ_i in equation 8 we can eliminate the last row of this equation and are left with two equations for the three unknowns per image point. As we have two images, we therefore have four equations for each of the three unknown 3d-coordinates per match. We can therefore solve for the unknown p by solving a linear system $\mathbf{Ap} = \mathbf{b}$, where $\mathbf{p}$ is a vector of all the unknown p's and $\mathbf{A}$ and $\mathbf{b}$ encode the equations (For details please refer to the code).

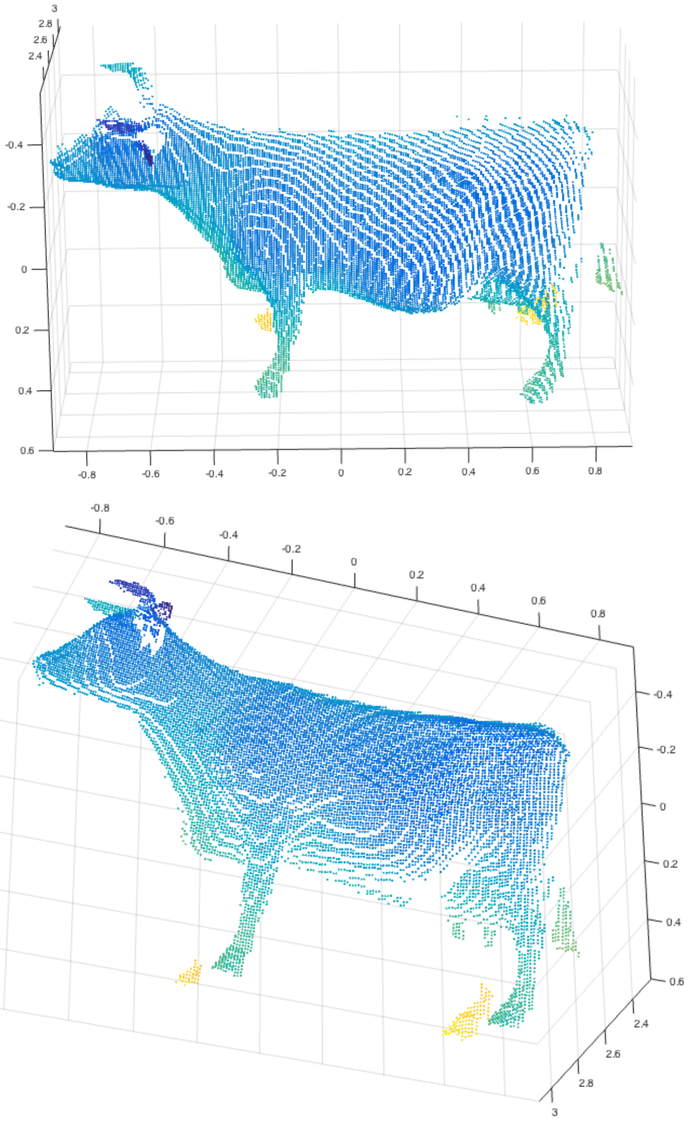

Fig. 5: Results of 3d-reconstruction.

Discussion Figure 5 shows the resulting 3d-point cloud. We observe that contrary to the first part of the assignment, where we only had few and noisy correspondences, the method works very well in the case of dense correspondences between the two images. This has been shown both by comparing to the ground truth transformation matrices and the visual results of the reconstruction.